LIBRARY
OF THE
UNIVERSITY
OF ILLINOIS

NOTICE: According to Sec. 19 (a) of the University *Statutes,* all books and other library materials acquired in any manner by the University belong to the University Library. When this item is no longer needed by the department, it should be returned to the Acquisition Department, University Library.

DIGITAL
COMPUTER LAB.

Allerton Conference on Circuit
and System Theory 1967

Proceedings

Digitized by the Internet Archive
in 2026 with funding from
University of Illinois Urbana-Champaign

https://archive.org/details/proceedings5196alle_0

7300-9
1.40

Proceedings

FIFTH ANNUAL ALLERTON CONFERENCE ON CIRCUIT AND SYSTEM THEORY

DIGITAL COMPUTER
LABORATORY
LIBRARY

October 4-6, 1967
University of Illinois
Allerton House, Monticello, Illinois
Sponsored by the
Department of Electrical Engineering and the
Coordinated Science Laboratory of the
University of Illinois, Urbana, Illinois and the
Circuit Theory and Automatic Control Groups of the
Institute of Electrical & Electronics Engineers

OCT 25 1968

DIGITAL COMPUTER
LABORATORY
LIBRARY

PROCEEDINGS
FIFTH ANNUAL ALLERTON CONFERENCE
ON CIRCUIT AND SYSTEM THEORY

EDITED BY

J. B. CRUZ, JR.
T. N. TRICK

Conference Co-Chairmen

Conference held

October 4, 5, 6, 1967
Allerton House
Monticello, Illinois

Sponsored by the

DEPARTMENT OF ELECTRICAL ENGINEERING
and the
COORDINATED SCIENCE LABORATORY
of the
UNIVERSITY OF ILLINOIS
Urbana, Illinois

and the

CIRCUIT THEORY AND AUTOMATIC CONTROL GROUPS
of the
INSTITUTE OF ELECTRICAL & ELECTRONICS ENGINEERS

FOREWORD

The Fifth Annual Allerton Conference on Circuit and System Theory was held at Allerton House from October 4-6, 1967. These Proceedings are a permanent record of the papers presented at that Conference. The papers describe new results in information processing, communication sciences, control systems, network theory, and computer-aided design.

In addition to the technical sessions, the remoteness of Allerton Park continues to provide a setting fruitful to the exchange of ideas throughout the Conference.

The Conference was co-sponsored by the Department of Electrical Engineering and the Coordinated Science Laboratory of the University of Illinois, and the IEEE Groups on Circuit Theory and Automatic Control. We acknowledge the encouragement and support of Professor W. Dale Compton, Director of the Coordinated Science Laboratory, and Professor E. C. Jordan, Head of the Department of Electrical Engineering at the University of Illinois. Special thanks are due Mrs. Divona Keel, who coordinated with extraordinary skill all Conference arrangements. Her assistance was indispensable in making the Conference successful. We thank Mr. Harold B. Lawler and Mrs. Rosa Townsend for their help with certain business aspects of the Conference. We are indebted to our colleagues for their vital assistance: Professors Robert T. Chien, Abraham Haddad, Peter Kokotović, Wataru Mayeda, Gernot Metze, William R. Perkins, Franco Preparata, Jiri Vlach and Nelson Wax.

J. B. Cruz, Jr.
T. N. Trick

TABLE OF CONTENTS

Foreword iii

GENERAL SESSION A 1
Chairman: J. B. Cruz, Jr., University of Illinois

FLOATING-POINT-ROUNDOFF ACCUMULATION IN DIGITAL-FILTER REALIZATIONS
I. W. Sandberg 1

A NEW COMPUTER METHOD OF SYSTEM ANALYSIS
Bharat Kinariwala 4

SENSITIVITY ANALYSIS 8
Chairman: M. E. Van Valkenburg, Princeton University

PERTURBATIONS OF EIGENVALUES AND EIGENVECTORS OF A NETWORK
C. A. Desoer 8

EIGENFUNCTION SENSITIVITY - AN APPROACH TO THE SOLUTION OF THE TOLERANCE PROBLEM
Devendar C. Reddy 12

SETS OF POSSIBLE STATES OF LINEAR SYSTEMS GIVEN PERTURBED OBSERVATIONS
H. S. Witsenhausen 19

DISTRIBUTED NETWORK ANALYSIS 28
Chairman: O. Wing, Columbia University

FURTHER STUDY OF THE EQUAL-STUB IMPEDANCE CLASS OF THE TRANSMISSION LINE FILTERS
Harvey J. Hindin 28

TRANSIENT ANALYSIS OF TRANSMISSION LINES THROUGH SAMPLING TECHNIQUES
G. Prada 29

A PHASE INTEGRAL ITERATIVE PROCEDURE FOR ANALYZING NONUNIFORM TRANSMISSION LINES
J. P. Cretella 39

FINITE STATE SYSTEMS 47
Chairman: C. L. Coates, University of Texas

A FORMULATION OF FUZZY AUTOMATA AND ITS APPLICATION AS A MODEL OF LEARNING SYSTEMS
W. G. Wee and K. S. Fu 47

STOCHASTIC FINITE-STATE SYSTEMS: FOUR MODELS AND THEIR RELATIONSHIPS
Celso de Renna e Souza and R. Jeffrey Leake 57

ON THE PERIOD AND CHARACTERISTIC EQUATION OF AN AUTONOMOUS F-CHANNEL PARALLEL LINEAR FEEDBACK SHIFT REGISTER
M. Y. Hsiao 67

STARTING STATE GENERATING FUNCTION FOR AUTONOMOUS P-ARY FEEDBACK SHIFT REGISTER
Frederic J. Mowle 77

SYSTEM THEORY 87
Chairman: T. N. Trick, University of Illinois

REAL TIME SCALING
A. Papoulis 87

A NEW TEST FOR RANK OF A CONTROLLABILITY-OBSERVABILITY MATRIX
Anthony Grammaticos and Marshall I. Epstein 97

INNER PRODUCT PROCEDURES FOR OBTAINING PROCESS REPRESENTATIONS
Lee E. Ostrander and Gerald H. Cohen 104

ON THE CONSTRUCTION OF STATE SPACES
James A. Resh 112

ACTIVE NETWORKS 119
Chairman: J. Vlach, A. S. Popov Research Institute, Prague and University of Illinois

TIME-SHARING ACTIVE RC FILTERS
Thomas G. Mancuso and Samuel C. Lee 119

ON HAYKIM'S METHOD FOR RC-ACTIVE FILTER SYNTHESIS
Joseph H. Wujek, Jr. 129

SYNTHESIS OF BASHKOW'S A MATRIX FOR THE GENERAL RC CASE
J. E. Steelman and S. Karni 139

MINIMIZATION OF TRAJECTORY SENSITIVITY 142
Chairman: P. Kokotovic, M. Pupin Institute, Belgrade and University of Illinois

AN OPTIMAL FIXED CONTROL STRUCTURE DESIGN WITH MINIMAL SENSITIVITY FOR A LARGE ELASTIC BOOSTER
Thomas Hendricks and Henry D'Angelo 142

OPTIMAL CONTROLLER DESIGN FOR MINIMUM TRAJECTORY SENSITIVITY
J.Y.S. Luh and E. R. Cross 152

TIME DOMAIN SENSITIVITY COMPARISONS
G. B. Lamont and S. J. Kahne 161

THE DESIGN OF OPTIMAL LOW SENSITIVITY LINEAR CONTROL SYSTEMS
Robert J. Dompe and Richard C. Dorf 170

MINIMUM SENSITIVITY DESIGN OF OPTIMAL LINEAR TRACKING SYSTEMS
M. K. Masten and B. F. Womack 181

CONTROL SYSTEMS 191
Chairman: W. R. Perkins, University of Illinois

ON IDENTIFICATION OF CONSTRAINED DYNAMIC SYSTEMS WITH HIGH DIMENSIONS
Masanao Aoki 191

ADAPTIVE CONTROL WITHOUT COMPLETE PLANT IDENTIFICATION
N. K. Sinha and M. J. Iqleem 201

A TECHNIQUE FOR THE SYNTHESIS OF DISCRETE ADAPTIVE CONTROL SYSTEMS
Vallobh Vimolvanich and Tien C. Hsia 211

APPLICATION OF MATHEMATICAL PROGRAMMING TO THE DESIGN OF OPTIMAL CONTROL SYSTEMS
B. C. Kuo and D. Tabak 213

DISCRETE SYSTEMS WITH TIME DELAY
Dong Hak Chyung 222

A METHOD FOR EVALUATING MULTIPLE MODES OF LINEAR SYSTEMS
C. F. Chen and D.W.C. Shen 228

STABILITY 229
Chairman: C. A. Desoer, University of California, Berkeley

OSCILLATIONS IN SYSTEMS WITH EVEN NONLINEARITIES
Samuel A. Musa 229

FINITE TIME STABILITY OF MULTI NON-LINEAR CONTROL SYSTEMS
A. N. Michel and S. H. Wu 230

A NEW METHOD FOR SOLVING $PA + A^TP = -D$
C. F. Chen and L. S. Shieh 231

PERIODICALLY TIME-VARYING SYSTEM: AN INSTABILITY CRITERION
Theodore A. Bickart 232

AN OFF-AXIS CIRCLE CRITERION FOR THE STABILITY OF FEEDBACK SYSTEMS WITH A MONOTONIC NONLINEARITY
Yo Sung Cho and Kumpati S. Narendra 238

STABILITY OF NONLINEAR TIME-VARYING FEEDBACK SYSTEMS
Yo Sung Cho and Kumpati S. Narendra 249

THE PROBLEM OF LUR'E IN SYSTEMS WITH TIME LAG
G. S. Tahim 259

DISTRIBUTED NETWORK THEORY 263
Chairman: B. Liu, Princeton University

DRIVING POINT IMMITTANCE SYNTHESIS USING EXPONENTIAL LINES
J. C. Giguere, M.N.S. Swamy and B. B. Bhattacharyya 263

THE REALIZABILITY AND DESIGN OF SYMMETRICAL n-PORT TRANSMISSION-LINE NETWORKS
Charles V. Shaffer 269

A GENERAL APPROXIMATION SYNTHESIS METHOD USING UNIFORM RC AND RCG LINES
Suleyman S. Penbeci and Samuel C. Lee 279

UNIFORM MULTI-LAYER RC DISTRIBUTED NETWORKS
F. Y. Chang and O. Wing 290

DISTRIBUTED AMPLIFICATION THEORY
Wai-Kai Chen 300

STOCHASTIC SYSTEMS 317
Chairman: F. Preparata, University of Illinois

ON A CLASS OF LINEAR STOCHASTIC DIFFERENTIAL GAMES
R. D. Behn and Y. C. Ho 317

PERFORMANCE MOMENT RECURSIONS, WITH APPLICATION TO EQUALIZER CONTROL LAWS
Michael K. Sain 327

LEARNING IN NONSTATIONARY ENVIRONMENT USING DYNAMIC STOCHASTIC APPROXIMATION
Y. T. Chien and K. S. Fu 337

ADAPTIVE STEP SIZE RANDOM SEARCH
M. A. Schumer and K. Steiglitz 346

STOCHASTIC GREEN'S FUNCTIONS AND THEIR RELATION TO THE RESOLVENT KERNELS OF INTEGRAL EQUATIONS
Leon H. Sibul 356

AN UPPER BOUND ON THE ZERO-CROSSING DISTRIBUTION
Nicholas A. Strakhov and Ludwik Kurz 364

STOCHASTIC APPROXIMATION ALGORITHMS FOR SYSTEM IDENTIFICATION, ESTIMATION, AND DECOMPOSITION OF MIXTURES
G. N. Saridis, K. S. Fu and Z. J. Nikolic 374

GENERAL SESSION B 384
Chairman: T. N. Trick, University of Illinois

ON-LINE CIRCUIT DESIGN: AN INTRODUCTION
Michael L. Dertouzos 384

PROPER FLOW PATHS IN VERTEX-WEIGHTED COMMUNICATION NETS
B. R. Myers and E. A. Davila 402

ON THE CHARACTERIZATION OF NONUNIFORM LOSSLESS TRANSMISSION LINES OF FINITE LENGTH
M. Wohlers 412

OPTIMAL ADAPTIVE CONTROL 418
Chairman: B. S. Morgan, Jr., F. J. Seiler Research Lab.

FEEDBACK CONTROL WITH MAGNITUDE CONSTRAINTS FOR SYSTEMS WITH UNKNOWN PARAMETERS
R. A. Werner 418

OBTAINING AND APPLICATIONS OF OPTIMAL SENSITIVITY FUNCTIONS
P. Kokotović, J. E. Heller and P. Sannuti 428

THREE SEGMENT METHOD IN THE SENSITIVITY DESIGN OF CONTROL SYSTEMS
Juraj Medanić 439

QUADRATIC INVARIANCE IN LINEAR SYSTEMS
N. H. McClamroch and J. K. Aggarwal 451

SENSITIVITY IN AN OPTIMAL CONTROL SYSTEM WITH DISTRIBUTED PARAMETERS
A. J. Koivuniemi and P. Kruh 459

NETWORK THEORY 460
Chairman: J. A. Resh, Michigan State University

ANOTHER EXTENSION OF THE DARLINGTON SYNTHESIS
T. R. Narasimhan and V. Ramachandran 460

DERIVED POSITIVE REAL FUNCTIONS AND MATRICES AND CONSEQUENT COEFFICIENT RELATIONS
N. K. Bose and F. M. Reza 461

ON THE RELATION BETWEEN A POSITIVE REAL FUNCTION AND A THREE-PORT MATRIX
Ezra Zeheb 470

COMMUNICATION SYSTEMS 482
Chairman: A. H. Haddad, University of Illinois

ON THE DETECTION CAPABILITY OF NONBINARY CODES USING PERMUTATION MAPPINGS
D. T. Tang and V. Y. Lum 482

DISTANCE MEASURES AND RELATED CRITERIA
Hisashi Kobayashi and John B. Thomas 491

SIGNAL DESIGN FROM A VARIATIONAL VIEWPOINT
R. L. Kashyap 501

ADAPTIVE RADAR DETECTION WITH REGULATED ERROR PROBABILITIES IN EXTENDED CLUTTER TARGET ENVIRONMENTS
Harold M. Finn 502

COMPUTER AIDED ANALYSIS AND DESIGN 513
Chairman: R. A. Rohrer, University of California, Berkeley

COMPUTER APPLICATIONS: CIRCUIT ANALYSIS
Donnamaie E. Meyerhoff 513

RAPID COMPUTER TIME RESPONSE CALCULATION FOR SYSTEMS WITH ARBITRARY INPUT SIGNALS
Christopher Pottle 523

HIGH-SPEED DIGITAL COMPUTER ANALYSIS OF NONLINEAR AND TIME-VARYING ELECTRONIC CIRCUITS
R. H. Allen and C. Pottle 534

TRIDIAGONALIZATION AND RELATED MATRIX AND NETWORK ALGORITHMS FOR A POLYNOMIAL SPACE
Thomas G. Marshall, Jr. 544

COMPUTER DESIGN OF MULTISTAGE TRANSISTOR BIAS CIRCUITS
M. A. Murray-Lasso and W. D. Baker 557

TWO CLASSES OF VARIATIONAL NETWORKS AND THEIR EXPLOITATION IN COMPUTER AIDED DESIGN
G. D. Hachtel 564

PREDICTING RESPONSE ERROR OF DIGITALLY SIMULATED TRANSFER FUNCTIONS
Joseph S. Rosko 572

TIME-VARYING AND NONLINEAR SYSTEMS 573
Chairman: B. C. Kuo, University of Illinois

EXTENSION OF THE VOLTERRA FUNCTIONAL TECHNIQUE TO THE ANALYSIS OF TIME-VARYING NONLINEAR SYSTEMS
James C. C. Su 573

"VOLTERRA SYSTEM FUNCTION" AND A CLASS OF NONLINEAR TIME-VARIANT SYSTEMS
S. Narayanan and A. Lavi 575

MULTI-POINT DESIGN OF TIME-VARYING FEEDBACK CONTROLLERS
D. M. Salmon and P. Kokotović 577

TRANSIENT RESPONSE OF NONLINEAR SYSTEMS
D. L. Schlicht and G. J. Thaler 586

PARAMETER ANALYSIS OF LINEAR TIME-VARYING SYSTEMS
R. Bittel and D. Siljak 596

CONTROL OF LINEAR TIME VARYING PLANTS WITH DIFFERENTIAL OPERATORS ACTING ON THE INPUT
R. V. Monopoli and C. R. Gitlitz 608

THE DESIGN OF SUBOPTIMAL LINEAR TIME-VARYING SYSTEMS
David L. Kleinman and Michael Athans 614

GRAPH THEORY I 627
Chairman: N. Wax, University of Illinois

GRAPHS WITH DEGENERACY IN BASIC CIRCUIT MATRICES
P. M. Lin 627

ON MAXIMALLY DISTINCT TREES
G. Kishi and Y. Kajitani 635

TOPOLOGICAL DEGREES OF FREEDOM OF ELECTRICAL NETWORKS
T. Tsuchiya, T. Ohtsuki, Y. Ishizaki, H. Watanabe,
Y. Kajitani and G. Kishi 644

TWO NEW CHARACTERIZATIONS OF PLANAR GRAPHS
Louis Weinberg 654

ELIMINATION OF SUBGRAPHS IN A SIGNAL FLOW GRAPH
W. Mayeda 671

ON THE RELATIONSHIP BETWEEN FLOW-GRAPHS AND THE TRAVELING SALESMAN PROBLEM
E. S. Davidson and G. Metze 683

GENERAL SESSION C 692
Chairman: W. R. Perkins, University of Illinois

IMPROVEMENT OF PHASE-LOCKED LOOP PERFORMANCE BY THE INTRODUCTION OF NONLINEARITIES
L. L. Cleland and B. J. Leon 692

STOCHASTIC APPROACHES TO SENSITIVITY 701
Chairman: J. B. Cruz, Jr., University of Illinois

A PROBABILISTIC APPROACH TO THE PARAMETER SENSITIVITY PROBLEM
B. Liu and M. E. Van Valkenburg 701

OPTIMAL CONTROL OF PLANTS WITH RANDOM SLOWLY-VARYING PARAMETERS
L. Benjamin Patrick and Henry D'Angelo 711

PERFORMANCE SENSITIVITY OF CONTROL SYSTEMS TO PARAMETER UNCERTAINTIES, DISTURBANCES, AND MEASUREMENT NOISE
Lewis Meier 721

GRAPH THEORY II 729
Chairman: G. Metze, University of Illinois

DOMINATING SET OF DIRECTED GRAPHS
Yumin Fu 729

DISTINGUISHING CHARACTERISTICS OF OPTIMUM TREE
Joseph E. Barbay and George W. Zobrist 730

GRAPH REALIZATION UTILIZING TREE, PATH, AND CIRCUIT LISTS
J. W. LaPatra and K. D. Willis 738

n-PORTS 739
Chairman: B. J. Leon, Purdue University

ON THE SYNTHESIS OF RLC n-PORTS
G. O. Martens 739

THE DETERMINATION OF PORT-TREE STRUCTURES OF TRANSFORMERLESS (n+1)-TERMINAL n-PORT NETWORKS FROM THEIR SHORT-CIRCUIT ADMITTANCE PARAMETER MATRICES
Dean J. Schmidlin and Frank J. Lupo 748

PERFECT MATCHING FOR INFINITELY ITERATED LOSSLESS STRUCTURES (With Applications to Practical Network Matching)
R. A. Rohrer 757

FILTERING 770
Chairman: John B. Thomas, Princeton University

OPTIMUM FILTERING WITH A CLASS OF NONLINEAR SYSTEMS
A. H. Haddad 770

SUBOPTIMAL DISCRETE TIME KALMAN FILTERING
David M. Layton 781

STUDIES IN DYNAMIC FILTERING
Willie L. McDaniel, Jr. 791

PATTERN RECOGNITION 792
Chairman: R. T. Chien, University of Illinois

STOCHASTIC APPROXIMATION NONPARAMETRIC TRAINING PROCEDURES FOR MULTI-CATEGORY PATTERN CLASSIFIERS
J. M. Schumpert and S. S. Yau 792

CONVERGENCE OF RISK IN ADAPTIVE PATTERN RECOGNITION PROCEDURES
S. C. Schwartz 800

EMPIRICAL DECOMPOSITION OF MIXTURES IN PATTERN RECOGNITIONS
T. L. Stewart and W. M. Hammond 808

COMPUTER RECOGNITION OF ALPHABETIC CHARACTERS THROUGH DECOMPOSITION INTO BASIC STROKES
Theodosios Pavlidis 810

FLOATING-POINT-ROUNDOFF ACCUMULATION IN DIGITAL-FILTER REALIZATIONS

I. W. SANDBERG
Bell Telephone Laboratories, Incorporated
Murray Hill, New Jersey

ABSTRACT*

The difference equation

$$w_n = \sum_{k=0}^{M} b_k x_{n-k} - \sum_{k=1}^{N} a_k w_{n-k} \quad , \quad n \geq N \tag{1}$$

with $M \leq N$ defines the behavior of a general time-invariant discrete filter which acts on an input sequence $x_0, x_1, x_2, \ldots$ to produce an output sequence $w_N, w_{N+1}, w_{N+2}, \ldots$ that depends on the starting values $w_0, w_1, \ldots, w_{N-1}$.

There is a vast literature concerned with techniques for designing discrete filters [i.e., for determining the a_k and the b_k in (1)] to meet specifications of various types (see, for example Refs. 1, 2 and 3), and a good deal of material is available on the subject of roundoff effects in fixed-point realizations of discrete filters (see, for instance, Refs. 4 and 5). In this paper we derive some bounds on a meaningful measure of the overall effect of roundoff errors for discrete filters realized as digital filters on a machine employing floating-point arithmetic operations. This type of realization, as opposed to the fixed-point kind, is of particular importance in connection with, for example, digital computer simulations of systems, as a result of the large dynamic range afforded by the floating-point mode.

There are basic differences concerning fixed-point and floating-point error estimation problems which stem from the fact that the modulus of every individual arithmetic error in the fixed-point mode is bounded by a constant determined by the machine, whereas the maximum modulus of the error in forming, for example, the floating-point sum of two floating-point numbers is proportional to the magnitude of the true sum. For this reason, the approach presented here, as well as the character of the results, are quite different from those of earlier writers concerned with fixed-point realizations.

It is assumed that:

(i) each machine number q is equal to sgn (q) $a\ 2^b$ in which the exponent b is an integer, and a, the mantissa, is a t-bit number contained in $[\frac{1}{2},1]$ or $[\frac{1}{2},1] \cup \{0\}$;

---

*The complete paper is contained in the Bell System Technical Journal, Vol. 46, October 1967.

(ii) the range of values of b is adequate to ensure that all computed numbers lie within the permissible range;

(iii) the machine operations of addition and multiplication are performed in accordance with standard rounding conventions (described, for example, by Wilkinson[6]); and

(iv) the coefficients a_k and b_k in (1) are machine numbers.

Several results are presented. In particular, for a large class of filters for which

$$D(z) \triangleq 1 + \sum_{k=1}^{N} a_k z^{-k} \neq 0 \text{ for all } |z| \geq 1,$$

it is proved that there is a function f(K) with $f(K) \to 0$ as $K \to \infty$ and a constant c, both dependent on the b_k, the a_k, the order in which the products on the right side of (1) are summed in the machine, and t, the number of bits allotted to the mantissa, such that

$$\langle e \rangle_K \leq c \langle y \rangle_K + f(K)$$

for all $K \geq N$, in which, with $\{y_n\}$ the computed output sequence of the realized filter,

$$\langle y \rangle_K = \left(\frac{1}{K+1} \sum_{n=0}^{K} |y_n|^2 \right)^{1/2}$$

and

$$\langle e \rangle_K = \left(\frac{1}{K+1} \sum_{n=0}^{K} |w_n - y_n|^2 \right)^{1/2}.$$

Bounds on f(K) and c are given that are not difficult to evaluate, and which, in many realistic cases, are informative. For example, for the second-order bandpass filter:

$$w_n = x_n - a_1 w_{n-1} - a_2 w_{n-2}, \quad n \geq 2 \qquad (2)$$

with a_1 and a_2 chosen so that its poles in the z-plane are at approximately ±45° and at distance approximately (but not less than) 0.001 from the unit circle, we find that c, an upper bound on the "asymptotic output error-to-signal ratio", is not greater than 0.58×10^{-4}, assuming that t = 27, that the terms on the right side of (2) are summed in the machine in the order indicated (from right to left), and that the x_n in (2) are machine numbers. If the x_n are not machine numbers, and hence must be quantized before processing, then $c \leq 0.76 \times 10^{-4}$.

In addition to error bounds, an inequality is derived which, if satisfied, rules out certain types of generally undesirable behavior such as self-sustained output limit cycles due to roundoff effects. This inequality is satisfied for the example described above.

REFERENCES

1. R. B. Blackman and J. W. Tukey, The Measurement of Power Spectra from the Point of View of Communication Engineering, Dover, 1959.

2. R. B. Blackman, Linear Data-Smoothing and Prediction in Theory and Practice, Addison-Wesley, Reading, Massachusetts, 1965.

3. J. F. Kaiser, Digital Filters, Chapter 7 of System Analysis by Digital Computer, edited by F. F. Kuo and J. F. Kaiser, John Wiley, New York, 1966.

4. W. R. Bennett, Spectra of Quantized Signals, BSTJ, Vol.27, pp. 446-472, July 1948.

5. J. B. Knowles and R. Edwards, Effects of a finite-word-length computer in a sampled-data feedback system, Proc. IEE (London) Vol. 112, pp. 1197-1207, June 1965.

6. J. H. Wilkinson, Rounding Errors in Algebraic Processes, Prentice Hall, Englewood Cliffs, New Jersey, 1963.

A NEW COMPUTER METHOD OF SYSTEM ANALYSIS

BHARAT KINARIWALA
Department of Electrical Engineering
University of Hawaii
Honolulu, Hawaii

ABSTRACT

A novel machine method of inverting linear operators (or, solving operator equations) is presented. The method is iterative with guaranteed convergence and it uses the original operator or a simple modification of it to test the sucessively more accurate solutions. It thus minimizes accumulation of round-off errors and provides the control of accuracy to the operator. For a parametric family of problems, it also promises a significant saving in overall computing time. In addition the method applies to all bounded linear Hilbert space operators for which no systematic inversion procedures exist.

The method has been tested on finite dimensional operators (i.e., matrices) and preliminary results indicate significant advantages over conventional Gauss and Gauss-Seidel methods, both with respect to accuracy and computation time.

I. INTRODUCTION

Ideal computing can be summed up in two words, speed and accuracy. Considerable advances have been made in the speed of computing but, quite often, the accuracy leaves much to be desired. Many numerical methods are developed to minimize the number of computations (i.e., time) whereas the degree of accuracy is an incidental with the operator or the user unable to exercise any control over it. Iterative methods, when applicable, have the virtue of providing desired accuracy but the convergence and the speed of convergence become the imponderables. Computing usually involves a parametric family of problems, e.g., frequency response for several sample values of frequency, search (systematic or trial and error) for optimum values of parameters, etc. The speed of convergence of iterative methods, if they converge at all, in such situations can be expected to be high since the final solution for a certain set of parameters becomes the initial guess for the set obtained under small changes. Since the initial guess will be close to the final solution for the new set relatively fewer iterations will yield the solution.

For a parametric family of problems, therefore, iterative methods with guaranteed convergence properties would represent ideal computing. This paper presents such a method for solving operator equations or inverting operators when these are bounded linear Hilbert Space operators. The discussion below is carried out with reference to finite dimensional operators (i.e., matrices) of interest, for example, in circuit analysis but the basic proof of the main theorem applies equally well to Hilbert Space operators.

II. SYSTEM ANALYSIS

The basic problem of System Analysis is concerned with the solution of the operator equation

$$AX = Y \tag{1}$$

where we are interested in the solution X where the operator A and the vector Y may undergo parametric changes. For example, in the frequency domain analysis of electronic circuits A would represent the matrix arising in loop, node, or state variable description of the network. The analysis problem is then to solve the set of simultaneous equations with complex coefficients repeatedly for different values of frequency. Moreover, this would usually be done for a variety of changes in network element value.

For infinite dimensional operators there are no known techniques for systematic solution of such problems. The main theorem in this paper gives one such method. For finite dimensional operators we review some of the most widely used methods [1] and show the potential of the new method in comparison.

III. GAUSS ELIMINATION METHOD

Consider a set of n linear algebraic equations in the n unknowns x_j:

$$\sum_{j=1}^{n} a_{ij}x_j = y_i, \quad i = 1, 2, \ldots, n \tag{2}$$

The equations are manipulated by repeated multiplication and subtraction to obtain equations of the form

$$\sum_{j=i}^{n} b_{ij}x_j = w_i, \quad i = 1, 2, \ldots, n \tag{3}$$

where b_{ij} and w_i are the modified constants. The equations are solved then by first obtaining x_n, then x_{n-1}, and so on.

The elimination method is efficient, but it suffers from accuracy problems because of the many subtractions involved and the consequent accumulation of round-off errors. It should also be apparent that the entire process must be repeated for every change in the parameters (e.g., frequency values, circuit elements, etc.). There are numerous ingenious variants of this method which provide greater efficiency in computing time, however the basic disadvantages remain the same.

Let us now examine a method which is not much used because it is successful only for a small class of problems. The purpose here is to point out certain desirable features of the method if and when it can be applied.

IV. GAUSS-SEIDEL ITERATION METHOD

In this method the equations (2) are rearranged to:

$$a_{ii}x_i = \sum_{j=1}^{n} (a_{ii}\delta_{ij} - a_{ij})x_j + y_i, \quad i = 1, 2, \ldots, n \tag{4}$$

where δ_{ij} = Kroneker delta.

The solution is obtained by iteration. An initial guess for the unknowns x_j is substituted in the right hand side of the equation to obtain the new set of values for x_j. These are then substituted in the right hand side and the process repeated. If the process converges, the solution can be obtained to the desired degree of accuracy since more iterations will give more accurate solutions. There is also the added advantage in that fewer iterations would be required for changes in parameters.

The method, in spite of the desirable features mentioned above, is rarely used because it converges for a very restricted class of problems. An added disadvantage is that the necessary and sufficient conditions are not expressible in a usable form for computational purposes. Consequently, it is not possible to know whether a given problem is solvable by this method prior to the application of the iterative process itself. The method is quite unreliable for general use.

There are certain features of this method which are desirable if they could be incorporated into a universally applicable method. The iterative nature of the method when it converges, provides both accuracy and efficiency when changes in parameters are made. Such a method is described below.

V. KINARIWALA ITERATION METHOD

The equations (2) are modified to:

$$k_i x_i = \sum_{j=1}^{n} (k_i \delta_{ij} - c_{ij}) x_j + a^*_{ji} y_j, \quad i = 1, 2, \ldots, n \qquad (5)$$

where

$$c_{ij} = \sum_{u=1}^{n} a^*_{ui} a_{uj}$$

$$k_i = \sum_{j=1}^{n} |c_{ij}|$$

$(\ )^*$ = complex conjugate.

The procedure is to choose a set of values for $X_1, X_2, \ldots, X_n$ and substitute these in the right hand side of the first equation and solve for a new value of X_1. Then, we substitute this value of X_1 and the original values of $X_2, \ldots, X_n$ in the right hand side of the second equation and obtain a new value of X_2. We continue this process until we have a new set of values for $X_1, X_2, \ldots, X_n$. We then start again with the first equation and find a new X_1, then a new X_2, etc. Each time through this process will always give a better set of values for $X_1, X_2, \ldots, X_n$.

The process is guaranteed to converge in all cases and the solution is obtained to a desired degree of accuracy by a comparison of the new and the old sets of values. Changes in parameters can be efficiently handled by starting the iterative process with the known solution for the old parameters.

A Fortran program for IBM 360/50 has been written to test the method and preliminary results indicate significant advantages over conventional Gauss and Gauss-Seidel Methods, both with respect to accuracy and computation time.

As an example, the method was tested for a family of ten problems with the initial parameters as follows:

$n = 20$

$a_{ij} = 10$ for $i = j$

$= -1$ for $i \neq j$

$y_1 = 29$

$y_i = 0$ for $i > 1$

Change of parameters: small changes (between 5% and 10%) in one of the a_{ij}'s.

For this family of problems the Gauss-Seidel Method does not converge. About 20% saving in time was obtained over the Gauss elimination method together with improved accuracy (by one significant digit).

It must be emphasized that the above results are for a specific example only. There is not enough experience with the method to make any general statements or conclusions. The example does point out the potential of the method and further results should be forthcoming soon.

The method described is a simple consequence of the theorem, statement and proof of which are outline below

THEOREM: Given an operator equation

$$AX = Y, \tag{6}$$

where A is a linear bounded invertible operator defined on a Hilbert Space, it is always possible to obtain the solution by iteration performed on a modified equation.

OUTLINE OF PROOF:

For an equation

$$X = TX + W \tag{7}$$

it can be shown that it is always possible to obtain a unique solution by iteration starting from any arbitrary initial guess provided the spectrum of T, S(T), is bounded by a constant less than 1, i.e.,

$$|S(T)| \leqq \alpha < 1. \tag{8}$$

If Equation (6) can be modified to

$$BAX = BY = W \tag{9}$$

such that

$$1 \geqq S(BA) \geqq \alpha_1 > 0 \tag{10}$$

then (9) can be rearranged to

$$X = (I-BA)X + W$$

or,

$$X = TX + W \tag{12}$$

with $|S(T)| \leqq \alpha < 1$.

Equation (12) can now be solved by iteration. All we need to do is ensure that (10) is satisfied. If B is chosen to be kA*, where k is an appropriate scalar and A* is the adjoint of A, it is easy to see that (10) is satisfied. One possible value of k is

$$\frac{1}{||A^*A||}\ .$$

Solution by iteration of (12) is thus assured.

The method described above is a simple corollary of this result.

ACKNOWLEDGEMENT

The author wishes to express his gratitude for generous support of the Air Force Office of Scientific Research under grant AF-AFOSR-1251-67.

REFERENCES

1. See any text on Numerical Methods, for example, R. Beckett and J. Hurt, Numerical Calculations and Algorithms, McGraw-Hill, New York, 1967; Ch. 4.

PERTURBATIONS OF EIGENVALUES AND EIGENVECTORS OF A NETWORK

C. A. DESOER
University of California, Berkeley, California

ABSTRACT

This paper considers a linear, time-invariant nondegenerate network whose equations are in the state form : $\dot{\underset{\sim}{x}} = \underset{\sim}{A}\underset{\sim}{x}$. For the case where the eigenvalues are distinct, first and second order perturbation formulas are developed for the perturbed system $\dot{\underset{\sim}{y}} = (\underset{\sim}{A} + \epsilon \hat{\underset{\sim}{A}})\underset{\sim}{y}$. These results generalize those of Papoulis (1966).

The calculation of perturbation due to element deviations is fundamental in engineering. The prediction of perturbations can be used effectively in design [1, 2]. Papoulis used node or loop analysis to calculate the first and second order perturbations of two-element kind networks [3]. For three element kind networks, he obtained the first order perturbation of eigenvalues. These calculations were based on the reality and symmetry of the usual matrices. We propose to calculate directly first and second order perturbations of eigenvectors and eigenvalues based on state equations. Our method is more general in that it does not make any assumption on the nature of the network elements. It simply assumes that the eigenvalues are distinct. The spirit of this paper is closely related to [4].

I. FORMULATION

The linear time-invariant network under consideration is assumed to have a state representation

$$\dot{\underset{\sim}{x}} = \underset{\sim}{A}\underset{\sim}{x} \tag{1}$$

where $\underset{\sim}{A}$ is a (not necessarily real) constant n × n matrix. $\underset{\sim}{A}$ is assumed to have n pairwise distinct eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. Call $\underset{\sim}{u}_1, \underset{\sim}{u}_2, \ldots, \underset{\sim}{u}_n$ the corresponding eigenvectors. Let $\underset{\sim}{v}_1, \underset{\sim}{v}_2, \ldots, \underset{\sim}{v}_n$ be the reciprocal basis, then

$$\left\langle \underset{\sim}{v}_i, \underset{\sim}{u}_j \right\rangle = \delta_{ij} \qquad i, j = 1, 2, \ldots, n. \tag{2}$$

where $\langle \cdot , \rangle$ denotes the complex scalar product. Suppose that due to changes in element values, temperature or ageing the given matrix $\underset{\sim}{A}$ becomes the perturbed matrix $\underset{\sim}{A} + \epsilon \hat{\underset{\sim}{A}}$ where ϵ is real and small; $\hat{\underset{\sim}{A}}$ is an arbitrary constant n × n matrix. Let the perturbed eigenvectors and eigenvalues be represented by the Taylor series

---

The research reported herein was supported in part by the National Science Foundation under Grant GK-716.

$$\underset{\sim}{u}_i(\epsilon) = \underset{\sim}{u}_i + \epsilon\, \underset{\sim}{u}_i^{(1)} + \epsilon^2\, \underset{\sim}{u}_i^{(2)} + \dots \tag{3}$$

$$\lambda_i(\epsilon) = \lambda_i + \epsilon\, \lambda_i^{(1)} + \epsilon^2 \lambda_i^{(2)} + \dots \tag{4}$$

Expressing the fact that these are eigenvectors and eigenvalues of $\underset{\sim}{A} + \epsilon\, \underset{\sim}{\hat{A}}$, expanding the products, and equating like powers of ϵ, we obtain

$$\underset{\sim}{A}\, \underset{\sim}{u}_i = \lambda_i \underset{\sim}{u}_i \tag{5}$$

$$\underset{\sim}{\hat{A}}\, \underset{\sim}{u}_i + \underset{\sim}{A}\, \underset{\sim}{u}_i^{(1)} = \lambda_i \underset{\sim}{u}_i^{(1)} + \lambda_i^{(1)} \underset{\sim}{u}_i \tag{6}$$

$$\underset{\sim}{\hat{A}}\, \underset{\sim}{u}_i^{(1)} + \underset{\sim}{A}\, \underset{\sim}{u}_i^{(2)} = \lambda_i \underset{\sim}{u}_i^{(2)} + \lambda_i^{(1)} \underset{\sim}{u}_i^{(1)} + \lambda_i^{(2)} \underset{\sim}{u}_i\,. \tag{7}$$

II. FIRST ORDER PERTURBATION

If we consider $\lambda_i^{(1)}$ as given, (6) is a linear equation in $\underset{\sim}{u}^{(1)}$. The matrix of this equation is $\underset{\sim}{A} - \lambda_i \underset{\sim}{I}$, a singular matrix; $\underset{\sim}{u}_i^{(1)}$ is thus determined to within an arbitrary vector in the one-dimensional null space of $\underset{\sim}{A} - \lambda_i \underset{\sim}{I}$. We take this arbitrary vector to be zero (for otherwise, it would amount to rescaling $\underset{\sim}{u}_i$ as a function of ϵ). Consequently, we put

$$\underset{\sim}{u}_i^{(1)} = \sum_{k=1}^{n}{}' \alpha_{ik} \underset{\sim}{u}_k \tag{8}$$

where the ' indicates that the term $k = i$ is omitted from the sum. Substitute (8) into (6) and multiply by $\underset{\sim}{v}_i$ on the left and also by $\underset{\sim}{v}_k$ on the left, we obtain

$$\lambda_i^{(1)} = \left\langle \underset{\sim}{v}_i, \underset{\sim}{\hat{A}}\, \underset{\sim}{u}_i \right\rangle \qquad (i = 1, 2, \dots, n) \tag{9}$$

$$\alpha_{ik} = \left\langle \underset{\sim}{v}_k, \underset{\sim}{\hat{A}}\, \underset{\sim}{u}_i \right\rangle / (\lambda_i - \lambda_k) \qquad i \neq k \tag{10}$$

Hence

$$\underset{\sim}{u}_i^{(1)} = \sum_{k=1}^{n}{}' \frac{\langle \underset{\sim}{v}_k, \underset{\sim}{\hat{A}}\, \underset{\sim}{u}_i \rangle}{\lambda_i - \lambda_k}\, \underset{\sim}{u}_k \qquad (i = 1, 2, \dots, n) \tag{11}$$

Eq. (9) and (11) give the first order perturbation for the eigenvalues and for the eigenvectors.

III. SECOND ORDER PERTURBATION

From (7)

$$(\underset{\sim}{A} - \lambda_i \underset{\sim}{I})\underset{\sim}{u}_i^{(2)} = - \hat{\underset{\sim}{A}}\underset{\sim}{u}_i^{(1)} + \lambda_i^{(1)} \underset{\sim}{u}_i^{(1)} + \lambda_i^{(2)} \underset{\sim}{u}_i \tag{12}$$

Noting that $\underset{\sim}{A} - \lambda_i \underset{\sim}{I}$ is singular, we again put

$$\underset{\sim}{u}_i^{(2)} = \sum_{k=1}^{n}{}' \beta_{ik} \underset{\sim}{u}_k \tag{13}$$

Substituting (13) into (12) and multiplying by $\underset{\sim}{v}_i$

$$\lambda_i^{(2)} = \left\langle \underset{\sim}{v}_i, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i^{(1)} \right\rangle \tag{14}$$

and by (11)

$$\lambda_i^{(2)} = \sum_{k=1}^{n}{}' \frac{\langle \underset{\sim}{v}_k, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i \rangle \langle \underset{\sim}{v}_i, \hat{\underset{\sim}{A}} \underset{\sim}{u}_k \rangle}{(\lambda_i - \lambda_k)} \tag{15}$$

Substituting (13) into (12) and multiplying by $\underset{\sim}{v}_k$ gives

$$(\lambda_k - \lambda_i)\beta_{ik} = - \left\langle \underset{\sim}{v}_k, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i^{(1)} \right\rangle + \alpha_{ik} \lambda_i^{(1)} \tag{16}$$

Hence

$$\beta_{ik} = - \frac{\langle \underset{\sim}{v}_i, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i \rangle \langle \underset{\sim}{v}_k, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i \rangle}{(\lambda_i - \lambda_k)^2} + \sum_{j=1}^{n}{}' \frac{\langle \underset{\sim}{v}_j, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i \rangle \langle \underset{\sim}{v}_k, \hat{\underset{\sim}{A}} \underset{\sim}{u}_j \rangle}{(\lambda_i - \lambda_j)(\lambda_i - \lambda_k)} \tag{17}$$

Hence by (13)

$$\underset{\sim}{u}_i^{(2)} = - \sum_{k=1}^{n}{}' \frac{\langle \underset{\sim}{v}_i, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i \rangle \langle \underset{\sim}{v}_k, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i \rangle}{(\lambda_i - \lambda_k)^2} \underset{\sim}{u}_k$$

$$+ \sum_{k=1}^{n}{}' \sum_{j=1}^{n}{}' \frac{\langle \underset{\sim}{v}_j, \hat{\underset{\sim}{A}} \underset{\sim}{u}_i \rangle \langle \underset{\sim}{v}_k, \hat{\underset{\sim}{A}} \underset{\sim}{u}_j \rangle}{(\lambda_i - \lambda_j)(\lambda_i - \lambda_k)} \underset{\sim}{u}_k \qquad (i = 1, 2, \ldots, n) \tag{18}$$

in each summation, the index may not be equal to i. The second order perturbation formulas are (14) and (18).

IV. CONCLUSIONS

1. The perturbation calculations for state equations is straightforward and does not require any assumption on the nature of the network elements.
2. The case where there are multiple eigenvalues is very complicated. As Papoulis points out, the leading term of the perturbation of eigenvalues is no longer proportional to ϵ.
3. Equation (4), (9) and (15) give the local behavior of the "eigenvalue locus." It is more general than the conventional root locus because here the coefficients of the characteristic polynomial are not necessarily linear in ϵ.
4. The model implicitely postulated by the form $A + \epsilon \hat{A}$ precludes the the consideration of singular perturbation. These would occur when additional stray capacitances and lead inductances would be taken into account.

REFERENCES

[1] S. Darlington, "Synthesis of Reactance 4-poles," Jour. Math and Phys. 18, p. 257-353, 1939.

[2] C. A. Desoer, "Network Design by First Order Predistorsion," IRE Trans., PGCT, CT [, 3, p. 167-170, Sept. 1957.

[3] A. Papoulis, "Perturbations of the Natural Frequencies and of the Eigenvectors of a Network," Trans. IEEE, CT-13, 2, p. 188-195, June 1966.

[4] C. A. Desoer, "Modes in Linear Circuits," IRE Trans., CT-7, p. 211-223, Sept. 1960.

EIGENFUNCTION SENSITIVITY - AN APPROACH TO THE SOLUTION OF THE TOLERANCE PROBLEM

DEVENDAR C. REDDY
UNIVERSITY OF ILLINOIS
Chicago, Illinois

ABSTRACT

The problem of relating the effect of small changes in the system parameters to the system performance [1] (transient response, etc.) in the case of linear, time-invariant, dynamic, systems can be effectively done using the concept of eigenfunction sensitivity. This very same concept can be used to advantage to solve the tolerance problem. The established techniques of linear programming prove a valuable tool in the solution of such a problem.

INTRODUCTION

Considerable amount of work has been done in the area of the study of effects of parameter variations on system performance i.e. the problem of analysis has been more or less successfully solved. A new angle can be added to the problem of analysis in the guise of the tolerance problem which is: Assuming that small changes do occur in the system parameters then how does one specify the tolerances on system components, so that the effects of parameter variations are, if not nullified, at least reduced? An approach to the solution of the above problem will be the object of this paper.

The concept of eigenvalue sensitivity is basic to the solution of such a problem. Using this concept bounds on the variations of the elements of the $\underline{A}$ matrix (which in effect specify the tolerances on system components) for a specific variation in the eigenvalues are obtained. The case of the single-input-single-output (single-input-multiple-output) system is thoroughly treated. Extension to the multivariable case should not prove to be very difficult.

The Tolerance Problem:

Consider the case of the linear, time-invariant, dynamic system defined by the set of equations

$$\dot{\underline{X}} = \underline{A_1 X} + \underline{BU} + \underline{NV} \tag{1}$$

$$\underline{U} = \underline{FY} \tag{2}$$

and

$$\underline{Y} = \underline{CX} \tag{3}$$

where the state vector $\underline{X}$ is an n-vector
the control vector $\underline{U}$ is an m-vector
the output vector $\underline{Y}$ is a q-vector
the disturbance input vector $\underline{V}$ is a k-vector
and $\underline{A_1}$, $\underline{B}$, $\underline{N}$, $\underline{F}$ and $\underline{C}$ are constant matrices of appropriate dimension.

Rewrite the above equations as

$$\dot{\underline{X}} = \underline{AX} + \underline{NV} \tag{4}$$

$$\underline{Y} = \underline{CX} \tag{5}$$

where $\underline{A} = \underline{A_1} + \underline{BFC}$

Let the elements of the $\underline{A}$ matrix be subjected to parameter variations. Let $\lambda_1, \lambda_2, \ldots \lambda_n$ be the 'n' distinct eigenvalues of the $\underline{A}$ matrix. Assuming that the variations are "small", it follows that

$$\delta\lambda_k = \sum_{i,j}^{n} \frac{\partial\lambda_k}{\partial a_{ij}} \, \delta a_{ij} \qquad (6)$$

$$i = 1, 2, \ldots n$$
$$j = 1, 2, \ldots n$$
$$k = 1, 2, \ldots n.$$

or

$$\underset{(n\mathrm{x}1)}{\begin{bmatrix} \delta\lambda_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \delta\lambda_n \end{bmatrix}} = \underset{(n\mathrm{x}n^2)}{\begin{bmatrix} \frac{\partial\lambda_1}{\partial a_{11}} & \cdots\cdots\cdots & \frac{\partial\lambda_1}{\partial a_{nn}} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \frac{\partial\lambda_n}{\partial a_{11}} & \cdots\cdots\cdots & \frac{\partial\lambda_n}{\partial a_{nn}} \end{bmatrix}} \underset{(n^2\mathrm{x}1).}{\begin{bmatrix} \delta a_{11} \\ \cdot \\ \cdot \\ \cdot \\ \delta a_{nn} \end{bmatrix}} \qquad (7)$$

It has been proved elsewhere [2] that if the only element that is varying is a_{ij} i.e. $a_{ij} = a_{ij} + \delta a_{ij}$ then

$$\delta\lambda_k = \frac{\Delta_{ij}(\lambda_k)}{\frac{d}{d\lambda}\left[\prod_{L=1}^{n} (\lambda-\lambda_L) \right] \Big|_{\lambda=\lambda_k.}} \cdot \delta a_{ij} \qquad (8)$$

where Δ_{ij} is the i,j^th^ cofactor of $\underline{A}$.

Next, consider the case when the $\underline{A}$ matrix is in the companion form i.e. let

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 \\ a_n & a_{n-1} & a_{n-2} & \cdot & & \cdot & & \cdot & a_1 \end{bmatrix} \qquad (9)$$

Next it can be proved that [3]

$$\underset{(n\mathrm{x}1)}{\begin{bmatrix} \delta\lambda_1 \\ \\ \delta\lambda_2 \\ \cdot \\ \cdot \\ \cdot \\ \\ \delta\lambda_n \end{bmatrix}} = \underset{(n\mathrm{x}n)}{\begin{bmatrix} \frac{\lambda_1^{n-1}}{S_1} & \frac{\lambda_1^{n-2}}{S_1} & \cdot\ \cdot\ \cdot\ \cdot & \frac{1}{S_1} \\ \frac{\lambda_2^{n-1}}{S_2} & \frac{\lambda_2^{n-2}}{S_2} & \cdot\ \cdot\ \cdot\ \cdot & \frac{1}{S_2} \\ \cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot & \cdot \\ \cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot & \cdot \\ \cdot & \cdot & \cdot\ \cdot\ \cdot\ \cdot & \cdot \\ \frac{\lambda_n^{n-1}}{S_n} & \frac{\lambda_n^{n-2}}{S_n} & \cdot\ \cdot\ \cdot\ \cdot & \frac{1}{S_n} \end{bmatrix}} \underset{(n\mathrm{x}1)}{\begin{bmatrix} \delta a_1 \\ \\ \delta a_2 \\ \cdot \\ \cdot \\ \cdot \\ \\ \delta a_n \end{bmatrix}} \qquad (10)$$

where $S_i = \langle \underline{X}_i, \underline{Y}_i \rangle$ (inner product of $\underline{X}_i$ and $\underline{Y}_i$)
and $\underline{X}_i$ is the right-hand eigenvector corresponding to the eigenvalue,
$\underline{Y}_i$ is the left-hand eigenvector corresponding to the eigenvalue.

It can be deduced from Eq. (10) that for non-zero S_i, $i = 1, 2, \ldots n$ the invariance of all the eigenvalues $\lambda_1, \lambda_2, \ldots\ldots \lambda_n$ is impossible.

Proof: Invariance of all eigenvalues implies that $\delta\lambda_1 = \delta\lambda_2 = \ldots = \delta\lambda_n = 0$ which implies that

$$\begin{array}{lcl} \frac{1}{S_1} \{ \lambda_1^{n-1}\delta a_1 + \lambda_1^{n-2}\delta a_2 + \ldots + 1\delta a_n \} & = & 0 \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \frac{1}{S_n} \{ \lambda_n^{n-1}\delta a_1 + \lambda_n^{n-2}\delta a_2 + \ldots + 1\delta a_n \} & = & 0 \end{array} \qquad (11)$$

i.e.

$$\begin{bmatrix} \lambda_1^{n-1} & \lambda_1^{n-2} & \cdots\cdots & 1 \\ \lambda_2^{n-1} & \lambda_2^{n-2} & \cdots\cdots & 1 \\ \cdot & \cdot & \cdots\cdots & \\ \cdot & \cdot & \cdots\cdots & \\ \cdot & \cdot & \cdots\cdots & \\ \lambda_n^{n-1} & \lambda_n^{n-2} & & \end{bmatrix} \begin{bmatrix} \delta a_1 \\ \delta a_2 \\ \cdot \\ \cdot \\ \cdot \\ \delta a_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \qquad (12)$$

Recognize immediately from Eq. (12) that the coefficient matrix is a Vandermonde matrix and that its rank is 'n'. Hence invariance of all the eigenvalues is impossible.

NOTE: Eq. (12) can be satisfied if and only if $\delta a_1 = \delta a_2 = \ldots = \delta a_n = 0$ which contradicts the hypothesis. Thus, if invariance of all the eigenvalues is not possible, then would it be possible to choose the tolerances $\delta a_1, \delta a_2, \ldots \delta a_n$ such that the sensitivity of all the eigenvalues is less than a specified value? The answer to this question will be the concern in the ensuing discussion.

Observe from Eq. (10) that $\delta\lambda_1, \delta\lambda_2, \ldots . \delta\lambda_n$ are linear in the variations $\delta a_1, \delta a_2, \ldots . \delta a_n$.

Recall the general linear programming problem which is

Minimize $\underline{CX}$ subject to the linear constraints $\underline{AX} \geq \underline{R}$ and $X \geq 0$ or

Maximize $\underline{R}^T\underline{Z}$ subject to the linear constraints $\underline{A}^T\underline{Z} \leq \underline{C}^T$ and $Z \geq 0$

where $\underline{A}$ is an mxn matrix
$\underline{X}$ is an nx1 column vector
$\underline{R}$ is an mx1 column vector
$\underline{Z}$ is an mx1 column vector
and $\underline{C}$ is an 1xn row vector.

Next re-cast the tolerance problem into the familiar linear programming problem. Thus

Maximize (corresponds to the worst case tolerances)

$$[\,|k_1| \quad |k_2| \quad \cdots \quad |k_n|\,] \begin{bmatrix} |\delta a_1| \\ |\delta a_2| \\ \\ |\delta a_n| \end{bmatrix} \tag{13}$$

subject to the linear constraints

$$\begin{bmatrix} \left|\frac{\partial\lambda_1}{\partial a_1}\right| & \left|\frac{\partial\lambda_1}{\partial a_2}\right| & \cdots\cdot & \left|\frac{\partial\lambda_1}{\partial a_n}\right| \\ \left|\frac{\partial\lambda_2}{\partial a_1}\right| & \left|\frac{\partial\lambda_2}{\partial a_2}\right| & \cdots\cdot & \left|\frac{\partial\lambda_2}{\partial a_n}\right| \\ \cdot & \cdot & \cdots\cdot & \cdot \\ \left|\frac{\partial\lambda_n}{\partial a_1}\right| & \left|\frac{\partial\lambda_n}{\partial a_2}\right| & \cdots\cdot & \left|\frac{\partial\lambda_n}{\partial a_n}\right| \end{bmatrix} \begin{bmatrix} |\delta a_1| \\ |\delta a_n| \\ \cdot \\ \cdot \\ |\delta a_n| \end{bmatrix} \leq \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \cdot \\ \cdot \\ \varepsilon_n \end{bmatrix} \tag{14}$$

and

$$\begin{bmatrix} |\delta a_1| \\ |\delta a_2| \\ \cdot \\ \cdot \\ |\delta a_n| \end{bmatrix} \geq 0 \tag{15}$$

where $k_1, k_2, \ldots k_n$ are the weighting factors for the tolerances $|\delta a_1|$, $|\delta a_2|, \ldots\ldots |\delta a_n|$, and $\varepsilon_1, \varepsilon_2, \ldots \varepsilon_n$ are in effect the regions of maximum tolerance of the eigenvalues in the s-plane. Having thus cast the problem into a general linear programming problem, the powerful Simplex method could be used to determine the worst case tolerances $|\delta a_i|$, $i = 1, 2, \ldots n$ for specified variations in the eigenvalues.

ILLUSTRATIVE EXAMPLE

Consider a linear, time-invariant system defined by the matrix differential equation

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{16}$$

The nominal values of a_2 and a_1 are 7 and 10 respectively. It is required to find the tolerances $|\delta a_2|$ and $|\delta a_1|$ such that the variations of the eigenvalues in the s-plane λ_1 and λ_2, namely $|\delta\lambda_1|$ and $|\delta\lambda_2|$, are less than or equal to 10% and 9% of their nominal values: let the weighting factors on the tolerances $|\delta a_1|$ and $|\delta a_2|$ be 1 and 3 respectively.

Solution: Since the problem involves a $2^{\underline{nd}}$ order system we shall use geometric method to solve the problem. Of course, for systems order higher than 3 the Simplex method would have to be used. As the upper bound on $|\delta\lambda_1|$ and $|\delta\lambda_2|$ is specified, it follows that

$$\left|\frac{\partial\lambda_1}{\partial a_1}\right| \; |\delta a_1| + \left|\frac{\partial\lambda_1}{\partial a_2}\right| \; |\delta a_2| \leq 0.2 \tag{17}$$

$$\left|\frac{\partial\lambda_2}{\partial a_1}\right| \; |\delta a_1| + \left|\frac{\partial\lambda_2}{\partial a_2}\right| \; |\delta a_2| \leq 0.45 \tag{18}$$

As the Eqs. (17) and (18) are linear, the maximum of $k_1 \; |\delta a_1| + k_2 \; |\delta a_2| = f$ occurs at the extreme point [see figure 1].

Substitution of the various quantities in the above equations yields

$$\frac{2}{3}\,|\delta a_1| + \frac{1}{3}\,|\delta a_2| \leq 0.2 \tag{19}$$

$$\frac{5}{3}\,|\delta a_1| + \frac{1}{3}\,|\delta a_2| \leq .45 \tag{20}$$

The maximum value of f occurs at the point $|\delta a_1| = .25$ and $|\delta a_2| = 0.1$ (see figure 1).

Discussion

It was pointed out earlier that invariance of all the eigenvalues was not possible for finite variations in $a_1, a_2, \ldots a_n$. However, invariance of some, not all, of the eigenvalues could be achieved but this would require that the some of the tolerances be specified a-priori. For instance, consider the case of a 3rd order single-input-single-output system with three distinct eigenvalues. If, say, the invariance of the eigenvalues λ_1 and λ_2 is desired, then it can be easily shown that

$$\delta a_1 = \frac{\delta a_3}{\lambda_1 \lambda_2} \quad \text{and} \quad \delta a_2 = -\frac{(\lambda_1 + \lambda_2)}{\lambda_1 \lambda_2}\,\delta a_3.$$

Thus a unique solution for δa_1 and δa_2 will exist only if δa_3 is specified uniquely.

Conclusions

The concept of eigenfunction sensitivity has been further utilized to determine the tolerances on the coefficients of the characteristic polynomial of a single-input-single-output, linear, time-invariant system. The tolerances mentioned above, in effect, specify the tolerances on the system components used.

The problem of determining the tolerance-bounds has been reduced to a general linear programming problem, so that the powerful Simplex method could be used to arrive at the results.

REFERENCES

[1] Reddy, D. C., "Parameter Sensitivity of Linear, Time-invariant, Dynamic Systems", Research Report R 315, Coordinated Science Laboratory, University of Illinois, Urbana, Illinois, Sept. 1966.

[2] Laughton, M. A., "Sensitivity in Dynamical System Analysis", Journal of Electronics and Control, Nov. 1964.

[3] Reddy, D. C., "Evaluation of the Sensitivity of an eigenvalue", IEEE Transaction on Automatic Control, (to appear as correspondence in the December 1967 issue).

[4] Gass, S. I., "Linear Programming", Methods and Applications, McGraw-Hill Book Company (1951).

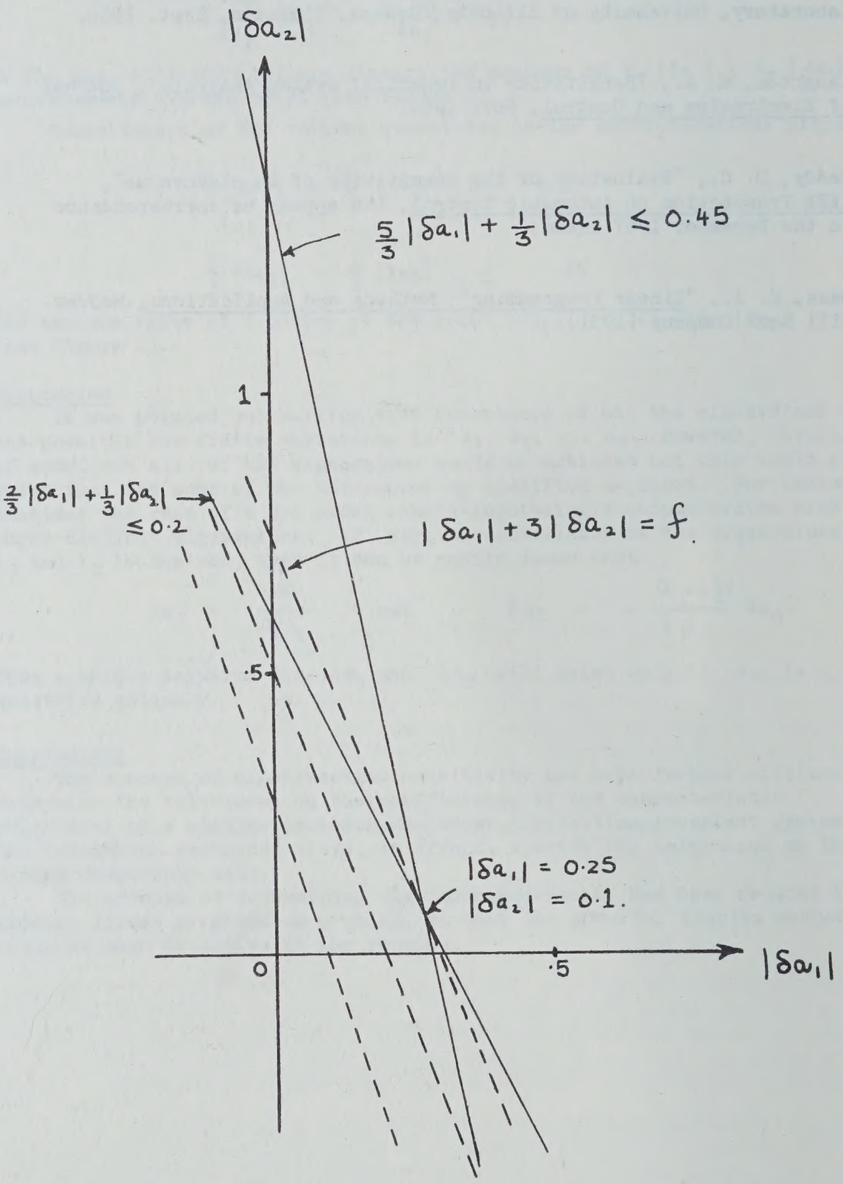

$|\delta a_2|$
$\frac{5}{3}|\delta a_1| + \frac{1}{3}|\delta a_2| \leq 0.45$
1
$\frac{2}{3}|\delta a_1| + \frac{1}{3}|\delta a_2| \leq 0.2$
$|\delta a_1| + 3|\delta a_2| = f.$
.5
$|\delta a_1| = 0.25$
$|\delta a_2| = 0.1.$
0
.5
$|\delta a_1|$

Figure 1

SETS OF POSSIBLE STATES OF LINEAR SYSTEMS GIVEN PERTURBED OBSERVATIONS

H. S. WITSENHAUSEN
Bell Telephone Laboratories, Incorporated
Murray Hill, New Jersey

ABSTRACT

The state equation $x_n = A_n x_{n-1} + B_n w_n$ and the output equation $y_n = C_n x_n + D_n w_n$ relate the finite-dimensional vectors x_n, y_n, w_n. The initial state x_0 is known to belong to a set X_0 and each w_n to a set W_n, where $X_0, W_1, \ldots, W_N$ are compact and convex. Define $S_n(y_1, \ldots, y_n)$ as the set of all possible values of x_n compatible with observation of outputs $y_1, \ldots, y_n$; it is compact and convex. S_n plays the same role as the a-posteriori distribution in the stochastic case and is determined recursively by $S_{n-1}(y_1, \ldots, y_{n-1})$ and y_n. The sets involved are completely characterized by their support functions. The law of evolution of the support function of S_n is established. Some special cases and applications are pointed out.

1. INTRODUCTION

Problems of stochastic state estimation are usually solved in two steps. The first step is to find the a-posteriori distribution of the state. This is usually done recursively. The second step is to find an estimate for which the expectation of the measure of error, under the a-posteriori distribution, is minimized. For instance, with quadratic criteria the optimal estimate is the a-posteriori mean.

If no probabilities are known but bounds are given for the uncertain variables, then the information available from the data can be stated in terms of the set of possible values of the state. Determination of this set constitutes the first step. The second step is to find an estimate for which the maximum of the error measure over that set is minimized. This is a minimax estimate, where the operations min and max are carried out in a specific order.

Such an approach differs from statistical decision theory in that no randomization of the decision (selection of the estimate) is allowed.

The first step then requires manipulation of sets, a major difficulty. For linear systems with perturbations constrained to compact convex sets all sets of interest are characterized by their support functions, which are a little easier to manipulate. The minimization in the second step can also be described in terms of support functions.

Finally, if a linear differential system is sampled at discrete times, then the effective constraint sets are reachable sets and the support function of these are more easily accessible than alternative descriptions.

2. PROBLEM STATEMENT

Let $L_x(n)$, $L_y(n)$, $L_w(n)$ be real linear spaces of finite dimensions $d_x(n)$, $d_y(n)$, $d_w(n)$ respectively. The dual spaces will be denoted by $L_x^*(n)$, $L_y^*(n)$, $L_w^*(n)$.

Let

$$A_n : L_x(n-1) \rightarrow L_x(n),\ B_n : L_w(n) \rightarrow L_x(n),\ C_n : L_x(n) \rightarrow L_y(n),\ D_n : L_w(n) \rightarrow L_y(n)$$

be linear maps.

Then the equations

$$x_n = A_n x_{n-1} + B_n w_n \tag{1}$$

$$y_n = C_n x_n + D_n w_n \tag{2}$$

where $n = 1,\ldots,N$, determine $x_1,\ldots,x_m$, $y_1,\ldots,y_m$ uniquely when $x_0, w_1,\ldots,w_m$ are given, $1 \leq m \leq N$.

Now suppose that nonempty compact convex sets $X_0 \subset L_x(0)$, $W_1 \subset L_w(1),\ldots,W_N \subset L_w(N)$ are given.

Then when $y_1,\ldots,y_n$ have been observed and it is known that $x_0 \in X_0$, $w_1 \in W_1,\ldots,w_n \in W_n$ one can only conclude that the sequence $x_0,x_1,\ldots,x_n$ must belong to a particular set of sequences to satisfy equations (1) and (2), the observations and the constraints. In particular the vector x_n must belong to the set $S_n(y_1,\ldots,y_n)$ defined recursively by

$$S_n(y_1,\ldots,y_n) = \{x_n | (1),(2),\quad x_{n-1} \in S_{n-1}(y_1,\ldots,y_{n-1}),\quad w_n \in W_n\} \tag{3}$$

$$S_0 = X_0 \tag{4}$$

Just before the observation y_n is made x_n is known to belong to the set $S_n'(y_1,\ldots,y_{n-1})$ defined recursively by

$$S_n'(y_1,\ldots,y_{n-1}) = \{x_n | (1),\quad x_{n-1} \in S_{n-1}(y_1,\ldots,y_{n-1}),\quad w_n \in W_n\} \tag{5}$$

$$S_0' = S_0 = X_0 \tag{6}$$

and y_n in $L_y(n)$ is known to belong to the set

$$M_n(y_1,\ldots,y_{n-1}) = \{y_n \mid (1),(2),\ x_{n-1} \in S_{n-1}(y_1,\ldots,y_{n-1}),\ w_n \in W_n\} \tag{7}$$

More precisely the pair (x_n,y_n), element of the direct sum $L_x(n) \oplus L_y(n)$, is known to belong to the set

$$C_n(y_1,\ldots,y_{n-1}) = \{(x_n,y_n) \mid (1),(2),\ x_{n-1} \in S_{n-1}(y_1,\ldots,y_{n-1}),\ w_n \in W_n\} \tag{8}$$

The problem is to determine $S_n(y_1,\ldots,y_n)$ recursively from $S_{n-1}(y_1,\ldots,y_{n-1})$ and the new observation y_n.

3. PRIMAL RECURSION

Comparing (3) and (8) one has

$$S_n(y_1,\ldots,y_n) = \{x_n \mid (x_n,y_n) \in C_n(y_1,\ldots,y_{n-1})\} \tag{9}$$

Define the set

$$T_n(y_n) = \{(x,y_n) \mid x \in L_x(n)\} \tag{10}$$

a linear manifold in $L_x(n) \oplus L_y(n)$.

Denote by π_n the canonical projection of $L_x(n) \oplus L_y(n)$ onto $L_x(n)$. Then (9) can alternatively be written

$$S_n(y_1,\ldots,y_n) = \pi_n\Big(T_n(y_n) \cap C_n(y_1,\ldots,y_{n-1})\Big) \tag{11}$$

Substituting (2) into (1) yields

$$y_n = C_n A_n x_{n-1} + (C_n B_n + D_n) w_n \tag{12}$$

from which

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \begin{bmatrix} A_n \\ C_n A_n \end{bmatrix} x_{n-1} + \begin{bmatrix} B_n \\ C_n B_n + D_n \end{bmatrix} w_n \tag{13}$$

where, e.g., $\begin{bmatrix} A_n \\ C_n A_n \end{bmatrix}$ is a linear map of $L_x(n-1)$ into $L_x(n) \oplus L_y(n)$.

By (8) and (13)

$$C_n(y_1,\ldots,y_{n-1}) = \begin{bmatrix} A_n \\ C_n A_n \end{bmatrix} S_{n-1}(y_1,\ldots,y_{n-1}) + \begin{bmatrix} B_n \\ C_n B_n + D_n \end{bmatrix} W_n \tag{14}$$

Equations (11) and (14) constitute in principle a solution of the problem, the primal solution. The required operations are linear mappings of sets, addition of sets and intersection of sets. Since these operations preserve both convexity and compactness the sets considered above are convex, and, with the exception of $T_n(y_n)$, compact.

4. SUPPORT FUNCTIONS

If L is a finite-dimensional real linear space with dual (set of linear functions) L^* and if S is a set in L the support function $h_S : L^* \to [-\infty, +\infty]$ of S is defined by

$$h_S(p) = \sup\{\langle p, x\rangle \mid x \in S\}. \tag{15}$$

We recall briefly a few properties of support functions [1,5,8]

(P1) If S is empty $h_S(p) \equiv -\infty$, otherwise h_S never takes the value $-\infty$.

(P2) h_S is real valued if and only if S is nonempty and bounded.

(P3) h_S depends only upon the closed convex hull of S and it characterizes the latter uniquely. Hence, closed convex sets are precisely characterized by their support functions.

(P4) If S, T, U are sets in L and S = T + U, that is

$$S = \{x+y \mid x \in T, \quad y \in U\}$$

then $h_S = h_T + h_U$ pointwise.

(P5) If $A : L_1 \to L_2$ is a linear map then

$$h_{AS}(p) = h_S(A^*p) \tag{16}$$

where $A^* : L_2^* \to L_1^*$ is the map dual to A.

(P6) If S1,...,Sn are closed convex sets in L, at least one of which is compact, then the support function of the intersection S of these sets is given by

$$h_S(p) = \inf\left\{\sum_{i=1}^{n} h_{Si}(p_i) \,\middle|\, \sum_{i=1}^{n} p_i = p\right\} \tag{17}$$

If the first sum above is indeterminate as $\infty - \infty$ then the infimum is understood over the range of indetermination, hence is $-\infty$.

Alternatively h_S is the lower convex envelope of the minimum of the h_{S_i}.

5. DUAL RECURSION

Since duality distributes on direct sums, one has [2]

$$\left(L_x(n) \oplus L_y(n)\right)^* = L_x^*(n) \oplus L_y^*(n)$$

and the elements of this space can be denoted by pairs (p,q) with $p \in L_x^*(n)$, $q \in L_y^*(n)$ and $\langle (x,y),(p,q)\rangle = \langle x,p\rangle + \langle y,q\rangle$.

Then (14) implies, by P4 and P5,

$$h_{C_n}(p,q;\ y_1,\ldots,y_{n-1}) = h_{S_{n-1}}\left(A_n^* p + A_n^* C_n^* q;\ y_1,\ldots,y_{n-1}\right)$$

$$+ h_{W_n}\left(B_n^* p + \left(B_n^* C_n^* + D_n^*\right) q\right) \tag{18}$$

Now, by (10) and (15), one has

$$h_{T_n}(p,q;\ y_n) = \begin{cases} \langle q, y_n\rangle & \text{for } p = 0 \\ +\infty & \text{for } p \neq 0 \end{cases} \tag{19}$$

By (17) and (19)

$$h_{T_n \cap C_n}(p,q;\ y_1,\ldots,y_n)$$

$$= \inf\left\{h_{C_n}(p,q+\eta;\ y_1,\ldots,y_{n-1}) - \langle \eta, y_n\rangle \,\middle|\, \eta \in L_y^*(n)\right\} \tag{20}$$

The dual map π_n^* is the canonical projection of $L_x^*(n) \oplus L_y^*(n)$ onto $L_x^*(n)$. Hence by (11) and (16) one obtains the dual recursion formula for the sets S_n:

$$h_{S_n}(p; y_1,\dots,y_n) = \inf\left\{h_{C_n}(p,\eta; y_1,\dots,y_{n-1}) - \langle \eta, y_n \rangle \mid \eta \in L_y^*(n)\right\}$$

$$= \inf\left\{h_{S_{n-1}}\left(A_n^* p + A_n^* C_n^* q; y_1,\dots,y_{n-1}\right)\right.$$

$$\left. + h_{W_n}\left(B_n^* p + \left(B_n^* C_n^* + D_n^*\right) q\right) - \langle q, y_n \rangle \mid q \in L_y^*(n)\right\} \tag{21}$$

Note that, from (5),

$$h_{S_n'}(p; y_1,\dots,y_{n-1}) = h_{S_{n-1}}\left(A_n^* p; y_1,\dots,y_{n-1}\right) + h_{W_n}\left(B_n^* p\right) \tag{22}$$

and, since M_n is the projection of C_n on $L_y(n)$,

$$h_{M_n}(q; y_1,\dots,y_{n-1}) = h_{C_n}(0,q; y_1,\dots,y_{n-1})$$

$$= h_{S_{n-1}}\left(A_n^* C_n^* q; y_1,\dots,y_{n-1}\right)$$

$$+ h_{W_n}\left(B_n^* C_n^* q + D_n^* q\right) \tag{23}$$

6. PROPERTIES

(1) The infimum in (21) is usually a minimum. An example where it is not is obtained if C_n is a sphere and T_n a plane tangent to this sphere, note that the observation y_n is then an extreme point of M_n.

(2) The infimum in (21) is that of a convex function of q. This is a most favorable situation for minimum seeking algorithms.

(3) Assume $h_{S_{n-1}}$ is concave jointly in $y_1,\dots,y_{n-1}$. Then h_{S_n} is the infimum of functions jointly concave in $y_1,\dots,y_n$ and is therefore itself jointly concave in these variables. By induction h_{S_n}, convex in p for fixed $y_1,\dots,y_n$, is concave in $y_1,\dots,y_n$ for fixed p.

The same property follows at once for $h_{S'_n}$, h_{M_n} and h_{C_n}.

In primal terms $S_n(y_1,\dots,y_n)$ is a *concave array* (see, e.g., Eggleston [1], page 81) and so are $C_n(y_1,\dots,y_{n-1})$, $S'_n(y_1,\dots,y_{n-1})$ and $M_n(y_1,\dots,y_{n-1})$. In turn, that these are concave arrays can be seen from the fact that the intersections of a convex set with translates of a linear manifold form a concave array with respect to the translation.

Formula (21) may also be interpreted as saying that the function $h_S(p; y_1,\dots,y_{n-1},\cdot)$ is the negative of the Fenchel transform [5], [8] of the function $h_{C_n}(p,\cdot; y_1,\dots,y_{n-1})$.

7. SPECIAL CASES

Suppose

$$x_n = A_n x_{n-1} + B_n u_n \tag{24}$$

$$y_n = C_n x_n + D_n v_n \tag{25}$$

with u_n and v_n independently constrained to compact convex sets U_n, V_n.

Then (21) takes the form

$$h_{S_n}(p; y_1,\dots,y_n) = \inf\Big\{h_{S_{n-1}}\Big(A_n^* p + A_n^* C_n^* q; y_1,\dots,y_{n-1}\Big) + h_{U_n}\Big(B_n^* p + B_n^* C_n^* q\Big) + h_{V_n}\Big(D_n^* q\Big) - \langle q, y_n\rangle \,\Big|\, q \in L_y^*(n)\Big\} \tag{26}$$

In the absence of the dynamic noise u_n the term in h_{U_n} disappears. In the absence of observation noise the term in h_{V_n} disappears, though the set S_n may still have more than one point when C_n is not one-one.

If there are no observations at all then,

$$h_{S_n}(p) = h_{S_{n-1}}\Big(A_n^* p\Big) + h_{U_n}\Big(B_n^* p\Big) \tag{27}$$

describes the open-loop propagation of uncertainty. Equation (24) may be obtained from a continuous-time linear system (A integrable, φ continuous);

$$\dot{\xi}(t) = A(t)\xi(t) + \varphi(t,u(t)) \tag{28}$$

where u is constrained to the set of all measurable functions with range in a compact set Ω. The samples $x_n = \xi(t_n)$ are observed with perturbations as per (25). Then in (24) A_n corresponds to the transition matrix $\Phi(t_n,t_{n-1})$, B_n is the identity and U_n is the reachable set at $t = t_n$ for (28), given that $\xi(t_{n-1}) = 0$. This set is compact and convex [4] and its support function is given by, [8],

$$h_{U_n}(p) = \int_{t_{n-1}}^{t_n} \sigma(\Phi^*(t_n,t)p,t)dt \tag{29}$$

where $\sigma(\cdot,t)$ is the support function of $\varphi(\Omega,t)$.

8. APPLICATIONS AND ALGORITHMS

One situation which can be modeled as above is the tracking of evasive targets, as studied by Schweppe [6].

Schweppe proposes an algorithm approximating all sets by ellipsoids. This has the disadvantage that even when the given sets $X_0,W_1,\ldots,W_N$ are in fact ellipsoids the sets S_n are generally not, so that an error exists even then. Furthermore, the given sets may be quite far from ellipsoids.

For these reasons it may sometimes be preferable to approximate all sets by polyhedra (convex hulls of finite sets). The advantages are threefold:

(a) any compact convex set can be approximated as closely as desired,

(b) if the given sets are in fact polyhedra then so are the sets C_n, S_n, S_n' and M_n,

(c) the algorithm for polyhedra reduces to a finite linear programming problem, though possibly of considerable size.

The method of Gardner [3] may be considered a special case of such an approach.

9. CONCLUSION

The state estimation problem considered here is to the stochastic state estimation problem for linear systems as the minimax control problem with observation of the state is to the stochastic control problem with observation of the state. One might expect that for quadratic criteria the minimax control problem with perturbed observations reduces to the two separate problems, as in the stochastic case. In fact, all the precise statements of this conjecture which come to mind are unfortunately false.

A primal solution of the combined problem is given in principle by dynamic programming as described elsewhere [7]. But calculations of nontrivial size seem inescapable in any actual solution of the combined problem.

REFERENCES

1. EGGLESTON, H. C., Convexity, Cambridge Univ. Press, 1958, p. 81.
2. GREUB, W. H., Linear Algebra, Springer 1967, p. 73.
3. GARDNER, L. A., Jr., Quarterly Progress Report, Sperry Rand Research Center, February 1967.
4. NEUSTADT, L. W., "The Existence of Optimal Controls in the Absence of Convexity Conditions," Jl. Math. Anal. Appl. 7, 110-117, 1963.
5. KARLIN, S., Mathematical Methods and Theory in Games, Programming and Economics, Vol. I, Addison-Wesley 1959, p. 226.
6. SCHWEPPE, F. C., "Recursive State Estimation When Observation Errors and System Inputs are Bounded," Proc. National Electronics Conference, 1967, (to appear).
7. WITSENHAUSEN, H. S., "Minimax Control of Uncertain Systems," Report ESL-R-269, MIT, April 1966.
8. WITSENHAUSEN, H. S., "A Minimax Control Problem for Sampled Linear Systems," (to appear).

FURTHER STUDY OF THE EQUAL-STUB IMPEDANCE CLASS OF TEM TRANSMISSION LINE FILTERS

HARVEY J. HINDIN
Senior Staff Engineer
LEL, Div. of Varian Assoc.
Copiague, N.Y. 11726

ABSTRACT

In previous work [1], an insertion loss analysis and exact design procedure was presented for a class of equal-stub impedance transmission line filters. These filters consisted of either quarter-wave shorted-stub resonators or half-wave open stub resonators coupled by a quarter-wavelength of transmission line. All of the stubs had the same characteristic impedance, the value of which was determined by the filter response requirements. These filters were easy to design, inexpensive, and useful for many applications.

Based on this, further effort has been expended to include other filters of the same general type. In particular a theoretical analysis has been made of the insertion loss vs. frequency characteristics of the quarter-wave open-stub resonator filter and the half-wave shorted-stub resonator filter. Generalizations for arbitrary lengths have also been made.

Because of the periodicity of the "unit-cell" for each section of the N-resonator filter, transmission matrix techniques are used for the derivation of the theoretical characteristics which are functions of Tchebychev polynomials of the second kind. The normalized frequency variable used is easily related to Richard's Transformation and is shown because of its properties to be a logical choice. Graphs are presented for the insertion loss vs. normalized frequency with normalized characteristic impedance as a parameter for one through eight resonators enabling exact design on an insertion loss basis for any frequency and bandwidth. In addition, simplified equations are derived approximating the exact equation in specfic regions of interest for all N. These filters are useful as band-pass or band-stop filters and design examples showing practical utilization are given. Also to be presented are equations and curves for various other electrical characteristics of interest. Experimental filters utilizing the design technique presented will be discussed and comparisons between theoretical and experimental data presented.

General equations and properties for these filters are derived and discussed to show what may be expected from filters of this class. A comparison is made between the characteristics of the filters presented here and in [1] and various general conclusions are made. Comparisons are also made between these filters and the more conventional TEM types. Advantages and disadvantages will be pointed out.

[1]. H.J.Hindin and J.J.Taub "Insertion Loss Analysis and Exact Design of the Equal-Element Class of TEM Transmission Line Filters" University of Illinois Fourth Allerton Conference, October 1966. See also IEEE-PGMTT Volume MTT-16, No. 9, Sept. 1967.

TRANSIENT ANALYSIS OF TRANSMISSION LINES THROUGH SAMPLING TECHNIQUES

G. PRADA
IBM Components Division, East Fishkill Facility, Hopewell Junction, N. Y.

ABSTRACT

This paper describes a new and simple method to calculate the transient response in transmission lines loaded with arbitrary linear loads placed at arbitrary distances on the line. The new approach uses sampling techniques and takes advantage of the properties of the Z-transform. This method replaces the sophisticated and time-consuming numerical algorithms required by other methods by a simple process of multiplication of polynomials. The simplicity of this method permits fast and easy implementation of computer programs to solve a given problem.

INTRODUCTION

When a transmission line is used to carry a signal from a generator to a series of loads, multiple reflections occur which deteriorate the traveling signal,causing considerable delays from input to output and sometimes multiple switching of the logic circuits. In high speed applications, it is important to know how the input signal is distorted as it reaches the loads in order to formulate adequate wiring rules to keep total delays and reflections within desirable limits.

Previous efforts to find the transient response in multi-loaded transmission lines make use of three basic techniques: Fourier analysis,[1,2] Laplace Transform,[3] and Time Domain Analysis.[4] The first two techniques involve the process of inverting the transfer function from the respective transform into the time domain. These inversion techniques are usually time-consuming and require sophisticated computer programs. The Time Domain Analysis involves the solution of differential equations or convolution integrals requiring special and complex numerical techniques. Other (less general) approaches avoid lengthy calculations by placing severe restrictions upon the loads, obviously limiting their usefulness to a very small and specific number of cases.

This paper describes a new and simple method to calculate the transient response in transmission lines loaded with arbitrary linear loads placed at arbitrary distances on the line. The new approach uses sampling techniques and takes advantage of the properties of the Z-transform. This method replaces the sophisticated and time-consuming numerical algorithms required by other methods by a simple process of multiplication of polynomials. The simplicity of this method permits fast and easy implementation of computer programs to solve a given problem.

THEORETICAL REVIEW

Analysis of Continuous Systems

As a result of using sampling techniques, the Z-transform becomes the basic mathematical tool of this paper. One of the most important properties of this transform is the following: If the Z-transform F(z) of a function f(t) is expanded into a power series in z^{-1} by the simple process of long division or by any other method, the terms of the series $\Sigma c_n z^{-n}$ can be identified, term by term, to the values of the generating function at times which are integral multiples of the sampling interval T. In other words, $c_n = f(nT)$ for $n = 0, 1, 2, \ldots$.

This suggests that the inversion process from the Z-transform into the time domain can be carried out by simple arithmetic methods of multiplication and division of polynomials. This special feature of the method makes the time response analysis of sampled systems much easier and amenable than the analysis of continuous systems, in which integration methods are used. This paper attempts to use some of the properties of sampling for solving the transient response of multiloaded transmission line networks.

In order to do this, it is necessary to introduce two fictitious elements in series with the continuous system.[5] The two elements are, an ideal switch S which closes every T units of time, and a data hold B whose function is to reconstruct the sequence of impulses arriving at its input into a function, $r_a(t)$, which is a close approximation to the signal r(t) in the continuous model (Fig. 1). The accuracy of this approximation depends mainly on the sampling interval T and on the type of data hold used. In the limit, as the sampling interval goes to zero, the sampled system is equivalent to the continuous one. On the other hand, as T increases, the accuracy of the approximation decreases. The rate of change of the error with respect to T depends directly on the frequency content of the signals being sampled.

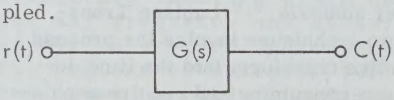


Fig. 1a. Continuous system.

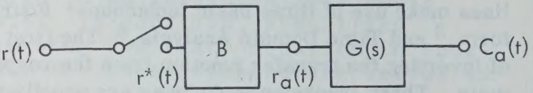


Fig. 1b. Sampled model of the system in Fig. 1a.

Since the function of the data hold is to reconstruct the sampled signal into its original continuous form, the accuracy of the approximation also depends on the type and sophistication of the data hold being used. For the purpose of this work, B has been chosen to be an element which converts a sequence of impulses into a continuous piece-wise linear function whose amplitude at times 0, T, 2T, ... is equal to the amplitude of the impulses at the same times. In sampled control systems, B is well known as the "Triangle Generator." Its transfer function is:[6]

$$B(s) = \frac{e^{sT}\left(1 - e^{-sT}\right)^2}{T\, s^2} \qquad (1)$$

Figure 2 provides a graphical representation of B(s). It is easy to observe that a decrease of the sampling interval would result in a better approximation to the input function r(t). It is also evident from Fig. 2a that (1) is physically unrealizable.

However, its use in this paper is just a mathematical artifice which permits the replacement of a continuous model by a sampled one.

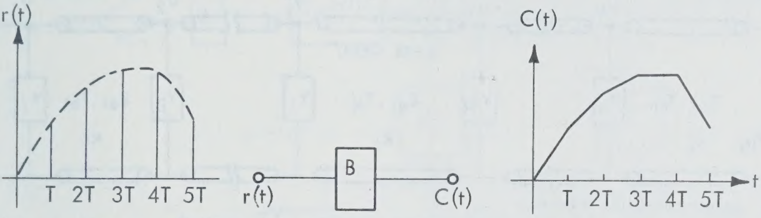


Fig. 2. Response of fictitious element.

With B(s) properly defined, the relation between the Z-transforms of the input and output in Fig. 1b can be written as:

$$C_a(z) = H(z)\,R(z) \tag{2}$$

where H(z) is the Z-transform of the function B(s) G(s), and R(z) is the Z-transform of the input. The inversion of C(z) into the time domain is then obtained by expressing both H(z) and R(z) as power series in z^{-1}, or

$$\sum_{0}^{\infty} c_k z^{-k} = \sum_{0}^{\infty} h_k z^{-k} \bullet \sum_{0}^{\infty} r_k z^{-k} \tag{3}$$

The coefficients C_k are then the values of the output function $C_a(t)$ at the times kT, k = 0, 1, 2... (Observe that without the use of the fictitious switch S, Eq. (2) would not be valid anymore.)

The advantage of the simulation of the continuous model by the sampled model is now clear. The process of finding the inverse Laplace transform of G(s) R(s) has been replaced by the much simpler one of multiplying the Z-transforms H(z) and R(z). For computing purposes, each of the series on the right hand side of (3) can be truncated after the m-th term,thus reducing this expressing to a simple product of polynomials whose result can also be truncated to give a polynomial of order m. The choice of m depends on the length of the desired time response. The truncation process provides results which are accurate up to the m-th term.

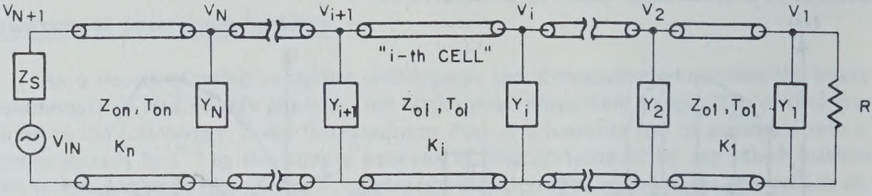
The technique of introducing a sampler followed by a data hold can be easily extended to the case of cascaded continuous systems. In this case it is necessary to replace each continuous element by its sampled model of Fig. 1b. As in the previous case, the inversion of the output function into the time domain can be reduced to the product of polynomials of order m.

TRANSMISSION LINE ANALYSIS

This section will attempt to use some of the properties of sampling for solving the transient response of multiloaded transmission line systems. The technique developed here is to replace the continuous model of a transmission line by a sampled model with fictitious samplers and data holds.

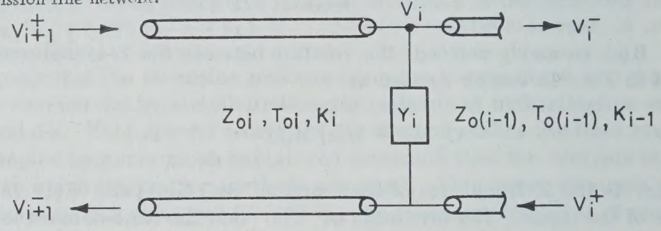
Consider the system of Fig. 3. For the purpose of mathematical treatment, this system can be considered made up of a cascade connection of N cells like the one shown in Fig. 4. Let Z_{oi} be the characteristic impedance of the i-th cell, Y_i its load admittance, T_{oi} the unloaded line delay, and K_i the attenuation factor.

For the moment it will be assumed that K_i is a constant, the condition for distortionless or low loss transmission lines.



V N+1
V N
V i+1
V i
V 2
V 1
"i-th CELL"
ZS
Zon, Ton
YN
Yi+1
Zoi, Toi
Yi
Y2
Zo1, To1
Y1
RT
VIN
Kn
Ki
K1

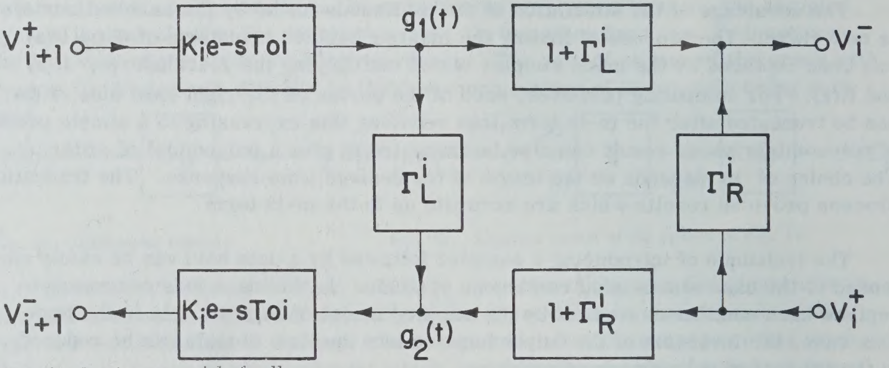
Fig. 3. Transmission line network.



V+ i+1
V i
V− i
Zoi, Toi, Ki
Yi
Zo(i-1), To(i-1), Ki-1
V− i+1
V+ i

Fig. 4. Transmission line cell.

The transmission line cell can be represented by the continuous model shown in Fig. 5. In this model Γ_L^i is the reflection coefficient that Y_i produces in a line of characteristic impedance Z_{oi} terminated in $Z_{o(i-1)}$, and it represents the fraction of the incident wave V_{i+1}^+ which is reflected back to the left port. Γ_R^i is the reflection coefficient that Y_i produces in a line of characteristic impedance $Z_{o(i-1)}$ terminated in Z_{oi}, and it represents the fraction of the incident wave V_i^+ which is reflected back to the right port.



V+ i+1
Kie−sToi
g1(t)
1+Γi L
V− i
Γi L
Γi R
V− i+1
Kie−sToi
g2(t)
1+Γi R
V+ i

Fig. 5. Continuous model of cell.

The functions $1+\Gamma_L^i$ and $1+\Gamma_R^i$ represent the fractions of the incident waves that are transmitted to the opposite node. An extensive analysis of transmission line flow graphs is found in reference 8.

The replacement of the continuous model by a sampled model can now be accomplished in a way similar to that described in the first section. However, in this case, two ideal switches and two data holds are required instead of one. The reason for the additional elements arises from the fact that the two incident waves ${}^+V_i$ and ${}^+V_{i+1}$ have to be sampled and reconstructed before applying them to the continuous elements Γ and $1+\Gamma$. See Fig. 6. The sampling of these two waves is performed by switches S1 and S2, respectively, and the reconstruction of the sam-

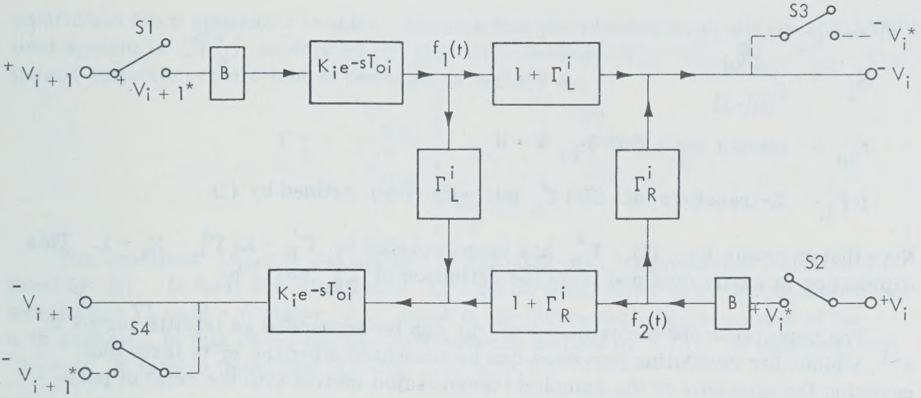


Fig. 6. Sampled model of cell.

pled waves by the same element B, defined before. It is then reasonable to assert that the continuous functions $f_1(t)$ and $f_2(t)$ in the sampled model are a close approximation to the functions $g_1(t)$ and $g_2(t)$, respectively, in the continuous model. The accuracy of the approximation depends mainly on the choice of the sampling interval.

At this point it should be pointed out that all switches close synchronously at equal intervals of time T. It is also important to realize that the sampling interval T should be a submultiple of the delay T_{oi}; and in the case of a multi-section transmission line, it should be a submultiple of the delays $T_{o1}, T_{o2}, \ldots, T_{on}$.

With the sampled model properly defined, it is possible to obtain a matrix relation that relates the incident and reflected sampled waves at the left port to the corresponding sampled waves at the right port. This relation is:†

$$\begin{bmatrix} {}^-V_i^* \\ {}^+V_i^* \end{bmatrix} = \begin{bmatrix} T_{11}^{i*} & T_{21}^{i*} \\ T_{21}^{i*} & T_{22}^{i*} \end{bmatrix} \begin{bmatrix} {}^+V_{i+1}^* \\ {}^-V_{i+1}^* \end{bmatrix} \tag{4}$$

The matrix parameters are a function of z, and will be called sampled or pulsed transmission parameters. The matrix will be referred to as the sampled transmission matrix. From Fig. 6 the matrix parameters are:

$$T_{12}^{i*} = \frac{K_i^{-1}}{X_i[1+B\Gamma_L^i(z)]\, z^{-N_{oi}}} \quad [X_i - 1 + X_i B\Gamma_L^i(z)]$$

$$T_{22}^{i*} = \frac{K_i^{-1}}{X_i[1+B\Gamma_L^i(z)]\, z^{-N_{oi}}} \tag{5}$$

$$T_{21}^{i*} = \frac{K_i}{X_i[1+B\Gamma_L^i(z)\, z^{-N_{oi}}} \quad [-B\Gamma_L^i(z)] \cdot z^{-2N_{oi}}$$

---

†Note the different nomenclature of the waves in the continuous and sampled models. In general, these waves are very close to each other, but not equal. The asterisk symbol denotes that the wave is sampled.

where

$$X_i = \frac{Z_{oi}}{Z_{0(i-1)}}$$

N_{oi} = integer such that $N_{oi} T = T_{oi}$

$B\Gamma_L^i$ = Z-transform of $B(s)\,\Gamma_L^i(s)$, with $B(s)$ defined by (1).

Note that to obtain Eq. (5), Γ_R^i has been replaced by $\Gamma_R^i = X_i \Gamma_L^i - X_i + 1$. This expression is easily obtained from the definition of Γ_L^i and Γ_R^i.

The numerator and denominator of (5) can be expressed as infinite series in z^{-1}, which, for computing purposes can be truncated after the m-th term thus covering the elements of the sampled transmission matrix into the ratio of polynomials of order m. Eq. (4) thus becomes:

$$\begin{bmatrix} {}^{-}V_i^* \\ \\ {}^{+}V_i^* \end{bmatrix} = \frac{1}{\sum\limits_{k=0}^{m} D_{i_k} z^{-k}} \begin{bmatrix} \sum\limits_{k=0}^{m} A_{i_k} z^{-k} & \sum\limits_{k=0}^{m} B_{i_k} z^{-k} \\ \sum\limits_{k=0}^{m} C_{i_k} z^{-k} & 1 \end{bmatrix} \begin{bmatrix} {}^{+}V_{i+1}^* \\ \\ {}^{-}V_{i+1}^* \end{bmatrix} \tag{6}$$

This matrix, obtained for the ith cell is particularly convenient in analyzing the system of Fig. 3. In fact, the incident and reflected sampled waves at the input of the (i-1)th cell are the reflected and incident sampled waves, respectively, at the output of the ith cell. Hence, the sampled transmission matrix of the overall system is the product of the sampled matrices $[T_i^*]$ of the individual cells:

$$[\hat{T}_N^*] = [T_1^*]\,[T_2^*]\,[T_3^*]\,.....\,[T_N^*] \tag{7}$$

and

$$\begin{bmatrix} {}^{-}V_1^* \\ \\ {}^{+}V_1^* \end{bmatrix} = [\hat{T}_N^*] \begin{bmatrix} {}^{+}V_{N+1}^* \\ \\ {}^{-}V_{N+1}^* \end{bmatrix} \tag{8}$$

The result expressed by (7) and (8) is very similar to that expressed by (3). The total sampled matrix relating to the input and output nodes has been reduced to the product of the sampled matrices of the individual cells. This means that the four elements of the matrix $[\hat{T}_N^*]$ can also be reduced to polynomials of order m, and are found by simple arithmetic means of multiplication of polynomials.

Pulsed Input Reflection Coefficient — Pulsed Transfer Coefficient

The two quantities of interest for the purpose of this work are the reflection coefficient at the input of a multiloaded transmission line system, and the transfer coefficient from generator to load. Assume that the transmission matrix of the N load system is $[\hat{T}^*_N]$ as defined by (7) with elements T^*_{11}, T^*_{12}, T^*_{21}, T^*_{22}. The pulsed reflection coefficient at the input is defined as:

$$\Gamma^*_N = \left. \frac{{}^{-}V^*_{N+1}}{{}^{+}V^*_{N+1}} \right|_{{}^{+}V^*_1 = 0} = \frac{-T^*_{21}}{T^*_{22}} \qquad (9)$$

The condition ${}^{+}V^*_1 = 0$ can always be satisfied with transmission matrices defined by (5). In fact, the admittance Y_1 can be considered to be the parallel of an admittance Y'_1 and a resistor Z_{on} equal to the characteristic impedance of the n-th section. In this case, the two port network defined by $[\hat{T}^*_N]$ is matched, and ${}^{+}V^*_1 = 0$. It is also found that $x_1 \cdot x_2 \ldots . x_n = 1$.

In a similar manner, the pulsed transfer function from node N+1 to node 1 is defined by:

$$H^{1*}_{N+1} = \left. \frac{{}^{-}V^*_1}{{}^{+}V^*_{N+1}} \right|_{{}^{+}V^*_1 = 0} = \frac{1}{T^*_{22}} \qquad (10)$$

Suppose now that the pulsed transfer coefficient from the input to an arbitrary node V_i is required. Since the analysis leading to Eq. (10) was general, it is expected that the same expression defines such transfer function, provided that the pulsed transmission matrix relating the node V^*_i and V^*_{N+1} is defined.

As before, call $[T^*_i]$ the pulsed transmission matrix of the i-th cell, and call $[T^*_{Ei}]$ the transmission matrix of the i-th cell when the admittance Y_i is replaced by the parallel of Y_i and the input admittance of the system consists of the cells (i-1), (i-2), ... , 2, 1.

It can be easily verified that the elements of the new matrix are given by Eq. (5), with $B\Gamma^i_L$ (z) replaced by: $B\Gamma^i_{LE}(z) = \Gamma^*_i \cdot z^{2N_{oi}}$, with Γ^*_i given by (9). Hence, the total pulsed transmission matrix relating the nodes N+1 and i can be expressed as

$$[T^{i*}_{N+1}] = [T^*_{Ei}] \cdot [T^*_{i+1}] \cdot [T^*_{i+2}] \ldots . [T^*_N] \qquad (11)$$

and the pulsed transfer function is given by the right hand side of (10).

Effect of the Generator Impedance

The transfer function H^{i*}_{N+1} from node N+1 to node i was defined without including the effect of the generator impedance $Z_g(s)$. It is easily verified that the pulsed transfer function from the generator to a node V_i is:

$$H_g^{i*} = \frac{BG^* \cdot H_{N+1}^{i*}}{1 - (1 - BG^*)\Gamma_N^*} \tag{12}$$

with Γ_N^* given by (9), and BG* being the Z-transform of the function

$$B(s)G(s) = \frac{Z_{on} \cdot B(s)}{Z_{on} + Z_g(s)} \tag{13}$$

A case which is not covered by (12) is the pulsed transfer function from the generator to node N-1. It is given by:

$$H_g^{N+1*} = \frac{BG^* (1 - \Gamma_N^*)}{1 - (1 - 2\ BG^*)\Gamma_N^*} \tag{14}$$

Time Response

From the very definition of Eq. (11), it is easy to realize that the pulsed transmission parameters of the matrix $[T_{N+1}^{i*}]$ can be reduced, after a truncation process, to a polynomials of order m in z^{-1}. The same result applies for Eq. (12), and the sampled time response V_i at the node i can then be expressed by:

$$\sum_{k=0}^{m} v_i(kT) z^{-k} = \sum_{k=0}^{m} h_k z^{-k} \bullet \sum_{k=0}^{m} V_g(kT) z^{-k} \tag{15}$$

where $V_g(t)$ is the sampled generator signal, and the h_k's are the coefficients of the power expansion of H_g^{i*}.
Note that the equal sign refers to the first m terms of the right hand side of (23) only.

Equation (15) provides the final result. It has been obtained after a simultaneous process of multiplication and truncations of polynomials of order m. No other algorithm or numerical technique is required.

EXAMPLE, RESULTS, ACCURACY, CONCLUSIONS

In order to verify the results, obtained in the previous sections, a computer program was written to calculate the response of the transmission line system of Fig. 7. For simplicity, the loads are assumed equal and equally spaced. The characteristic impedance of all sections is 50 ohms, and the total delay of each section is $T_o = 0.25$ ns.

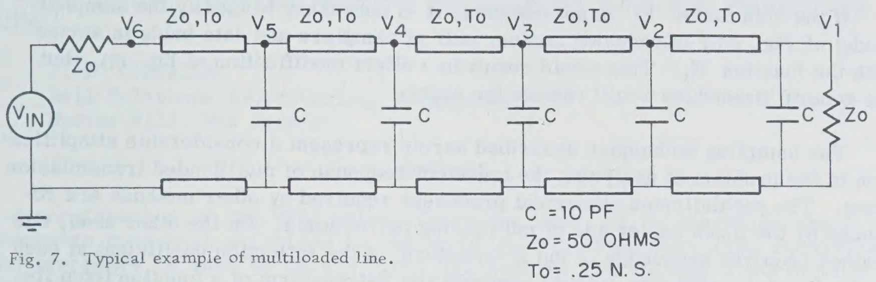


Fig. 7. Typical example of multiloaded line.

Figure 8 shows the input wave form, and the response of the system at the nodes V_1, and V_5. The same curves represent the responses obtained with two other computer programs that utilize some of the techniques discussed in previous paragraphs. The two programs are TLCAP which uses the differential equation approach, and Fourier which uses the Fourier analysis method. It was found that the difference in the results of the three methods is less than the accuracy of Fig. 8.

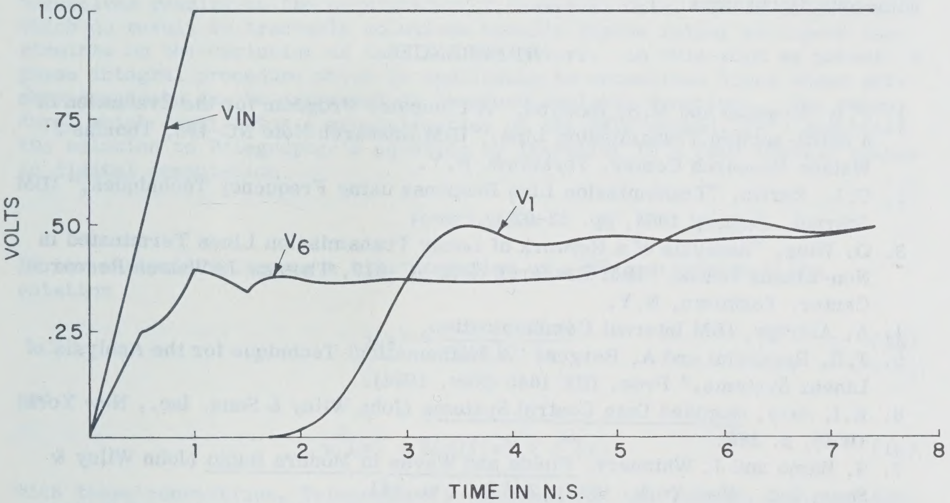


Fig. 8. Transient response of network in Fig. 7.

In order to have an estimate of the relative accuracy of the Z-transform method, the results were compared with the most accurate ones obtained with TLCAP. Two sampling intervals were used: T = 0.05 ns., and T = 0.25 ns. The first case resulted in an average error of 0.1%; the second case, in 1.5%. However, the computer time decreased from 13.50 seconds of IBM 7094 in the first case, to 0.88 seconds in the second. Computer times for TLCAP to obtain equivalent accuracies were 20.11 and 57.5 seconds, respectively. It is therefore apparent that with the proper choice of the sampling interval, the decrease in computation time is significant. On the other hand, the relation between the execution time and the length of the required time response is not linear. This can result in excessive amounts of computer time in cases in which a very long response output is required.

If the loss factor K_i is not constant, it is necessary to modify the sampled model of Fig. 6 by introducing another pair of samplers and data holds in series with the function K_i. This would result in a slight modification of Eq. (5), but the general procedure would remain the same.

The sampling techniques described herein represent a considerable simplification to the problem of analyzing the transient response of multiloaded transmission lines. The sophisticated numerical processes required by other methods are replaced by the much easier one of multiplying polynomials. On the other hand, our method requires knowledge of the Z-transform of the reflection coefficient at each load. Special techniques exist to generate the Z-transform of a function from its Laplace transform. The author suggests the use of the method described by Boxer and Thaler.[10]

ACKNOWLEDGMENTS

The author is specially indebted to J.P. Corrigan of IBM for his invaluable comments and criticisms, and to E. Lombrozo for his fruitful suggestions during numerous discussions.

REFERENCES

1. P.A. Brennan and M.S. Axelrod, "A Computer Program for the Evaluation of a Multi-section Transmission Line," IBM Research Note NC-498, Thomas J. Watson Research Center, Yorktown, N.Y.
2. C.L. Bertin, "Transmission Line Response using Frequency Techniques," IBM Journal, January 1964, pp. 52-63.
3. O. Wing, "Analysis of a Network of Lossy Transmission Lines Terminated in Non-Linear Loads," IBM Research Note NC-619, Thomas J. Watson Research Center, Yorktown, N.Y.
4. A. Aldrige, IBM Internal Communication.
5. J.R. Ragazzini and A. Bergen, "A Mathematical Technique for the Analysis of Linear Systems," Proc. IRE 1645 (Nov. 1954).
6. E.I. Jury, Sampled Data Control Systems (John Wiley & Sons, Inc., New York, 1958), p. 288.
7. S. Ramo and J. Whinnery, Fields and Waves in Modern Radio (John Wiley & Sons, Inc., New York, 1956), 2nd Ed., p. 461.
8. S.J. Mason and H.J. Zimmerman, Electronic Circuits, Signals, and Systems (John Wiley & Sons, Inc., New York, 1960), Ch. 4, 5, and 7.
9. E.I. Jury, op. cit. p. 405.
10. R. Boxer and S. Thaler, "A Simplified Method of Solving Linear and Non-Linear Systems," Proc. IRE, Jan. 1956, p. 89.

A PHASE INTEGRAL ITERATIVE PROCEDURE FOR ANALYZING NONUNIFORM TRANSMISSION LINES

J. P. CRETELLA
Bell Telephone Laboratories, Incorporated
Murray Hill, New Jersey

ABSTRACT

A generalization of the WKBJ method forms the basis of an iteration which generates numerical solutions to the nonuniform transmission line equations. The procedure is used to obtain plots delineating the voltage distribution and input impedance along exponentially tapered lines and Bessel lines. The results of the approximation compare very favorably with the exact solutions which are obtainable by standard methods of analysis for these particular lines.

INTRODUCTION

This study is concerned with a procedure which facilitates the extraction of numerical data from the nonuniform transmission line equations. Most analytical treatments of inhomogeneous lines result in solutions which do not lend themselves readily to the acquisition of numerical data. Those procedures which do result in tractable solutions usually impose rather stringent constraints on the variation of the line parameters. In this work we present a phase integral procedure which is applicable to nonuniform lines whose primary constants can be expressed in terms of analytic functions. Our procedure, which constitutes a generalization of the WKBJ method,[1,2] generates the solution to Telegrapher's equation by iteration in a form ideally suited to digital computation.

FORMULATION

In our analysis we assume a time variation of $e^{j\omega t}$ and introduce the notation

$$c(x) = \frac{1}{2Z(x)} \frac{dZ(x)}{dx} \tag{1a}$$

and

$$\gamma^2(x) = Z(x)Y(x) + c^2(x). \tag{1b}$$

With these conventions, Telegrapher's equation for the voltage distribution on the line assumes the form

$$\frac{d^2V(x)}{dx^2} - 2c(x)\frac{dV(x)}{dx} - [\gamma^2(x) - c^2(x)]V(x) = 0. \tag{2}$$

The boundary conditions which accompany this equation are

$$V(x_o) = V_L \tag{3a}$$

and

$$\left.\frac{dV(x)}{dx}\Big/V(x)\right|_{x=x_o} = -\frac{Z(x_o)}{Z_L}, \tag{3b}$$

where V_L and Z_L are the voltage and impedance, respectively, at the load end of the line, $x=x_o$. If we express (2) in terms of the logarithmic derivative of the voltage,

$$\rho(x) = - \frac{dV(x)}{dx} \Big/ V(x),$$

the result is a Ricatti differential equation with variable coefficients,

$$\frac{d\rho(x)}{dx} = \rho^2(x) + 2c(x)\rho(x) - [\gamma^2(x) - c^2(x)]. \tag{4}$$

The boundary condition imposed on $\rho(x)$ is

$$\rho(x_o) = - \frac{dV(x)}{dx} \Big/ V(x) \Bigg|_{x=x_o} = \frac{Z(x_o)}{Z_L} . \tag{5}$$

The assumption made in cases where the WKBJ method is applicable is that $\rho(x)$ varies slowly enough that (4) can be solved by an iterative process.[1] In our formulation we make no a priori assumptions regarding the degree of variation of $\rho(x)$. Instead, we observe that for the special case where the coefficients in (4) are constant,* e.g., $\gamma(x) = \gamma_o$ and $c(x) = c_o$, the solution is

$$\rho(x) = - \gamma_o \tanh\left[\gamma_o\left(x - \frac{1}{\gamma_o} \tanh^{-1} \frac{\rho(x_o) + c_o}{\gamma_o}\right)\right] - c_o. \tag{6}$$

This lends us a clue as to the functional form of the solution for the general case of variable coefficients. In fact, it suggests that the form of the solution might be

$$\rho(x) = - \gamma_o \tanh[\gamma_o \varphi(x)] - c(x), \tag{7}$$

where γ_o is a constant (perhaps the average value of $\gamma(x)$ over the entire length of line), and $\varphi(x)$ is a function to be determined. Substitution of (7) into (4) generates the differential equation

$$\frac{d\varphi(x)}{dx} = 1 + \frac{1}{\gamma_o^2}\left[\gamma^2(x) - \gamma_o^2 - \frac{dc(x)}{dx}\right]\cosh^2 \gamma_o \varphi(x) \tag{8}$$

for the unknown function $\varphi(x)$. While, in general, the solution to this equation cannot be obtained in terms of elementary quadratures, it forms the keystone for a scheme which allows us to obtain a piecewise approximation to $\varphi(x)$ over the entire length of line. Differentiating (8) repeatedly with respect to x, we observe that the nth derivative of $\varphi(x)$ is a function of its first n-1 derivatives. For example, the second derivative of $\varphi(x)$ is a function of $\varphi(x)$ and $d\varphi(x)/dx$,

$$\frac{d^2\varphi(x)}{dx^2} = - \frac{1}{\gamma_o^2}\left\{ \left[2\gamma(x) \frac{d\gamma(x)}{dx} - \frac{d^2c(x)}{dx^2}\right]\cosh^2 \gamma_o \varphi(x) \right.$$

$$\left. + \gamma_o \frac{d\varphi(x)}{dx}\left[\gamma^2(x) - \gamma_o^2 - \frac{dc}{dx}\right]\sinh 2\gamma_o \varphi(x)\right\}. \tag{9}$$

---

* Such is the case in uniform lines and in lines with an exponential taper.

Hence, if we can specify the value of $\varphi(x)$ at some point x_p, we can subsequently specify the values of all its derivatives at that point. Having these values, however, is equivalent to having the Taylor expansion of $\varphi(x)$ about x_p. This suggests a procedure for determining $\varphi(x)$ by the process of analytic continuation. The one piece of information needed to start the scheme is obtained at the load end of the line, $x = x_o$, where it is known that $\rho(x_o) = Z(x_o)/Z_L$, cf. (5). Using this in (7), we can solve for $\varphi(x_o)$,

$$\varphi(x_o) = - \frac{1}{\gamma_o} \tanh^{-1} \frac{\rho(x_o) + c(x_o)}{\gamma_o} . \tag{10}$$

Knowing $\varphi(x_o)$, we can evaluate the successive derivatives

$$\left. \frac{d^n \varphi(x)}{dx^n} \right|_{x=x_o} .$$

It then becomes possible to analytically continue $\varphi(x_o)$ to the point $x_o + \Delta x$ by using a truncated Taylor series. Having obtained $\varphi(x_o+\Delta x)$, we can then determine $\varphi(x_o+2\Delta x)$ and so on out to $\varphi(\ell)$, where $x = \ell$ is the sending end of the line. While this technique does not provide a closed form expression for $\varphi(x)$, observe how readily it lends itself to digital computation. A piecewise representation for $\varphi(x)$, over the interval $x_o \leq x \leq \ell$, can be obtained by simply repeating the steps above. At each stage of the iteration, we select the increment Δx small enough so that $\varphi_i(x)$ is adequately represented by the first two terms of its Taylor series, i.e.,

$$\varphi_i(x) = a_o^{(i)} + a_1^{(i)}(x-x_{i-1}) \qquad x_{i-1} \leq x \leq x_i \tag{11}$$

with

$$a_o^{(i)} = \varphi_{i-1}(x_{i-1}) \tag{12a}$$

and

$$a_1^{(i)} = \left. \frac{d\varphi(x)}{dx} \right|_{x=x_{i-1}} . \tag{12b}$$

In terms of (11), $\rho_i(x)$ becomes

$$\rho_i(x) = - \gamma_o \tanh\left[\gamma_o \left(a_o^{(i)} + a_1^{(i)}(x-x_{i-1}) \right) \right] - c(x) \tag{13}$$

over the interval $x_{i-1} \leq x \leq x_i$. Then, since the voltage at a point x_p is given by

$$V(x_p) = V(x_o) \exp \left[- \int_{x_o}^{x_p} \rho(x) dx \right] , \tag{14}$$

we have the representation

$$V(x_p) = V(x_o)\exp\left[\sum_{i=1}^{p}\int_{x_{i-1}}^{x_i}\gamma_o\tanh\gamma_o\left(a_o^{(i)} + a_1^{(i)}(x-x_{i-1})\right)dx\right]$$

$$\times\exp\left[\int_{x_o}^{x_p}c(x)dx\right]. \tag{15}$$

Performing the indicated integrations and rearranging the result, we obtain

$$V(x_p) = V(x_o)\sqrt{Z(x_p)/Z(x_o)}$$

$$\times\prod_{i=1}^{p}\left(\frac{\cosh\xi_1^{(i)}x_i}{\cosh\xi_1^{(i)}x_{i-1}}\;\frac{1+\tanh\xi_o^{(i)}\tanh\xi_1^{(i)}x_i}{1+\tanh\xi_o^{(i)}\tanh\xi_1^{(i)}x_{i-1}}\right)^{1/a_1^{(i)}}, \tag{16}$$

with

$$\xi_1^{(i)} = \gamma_o a_1^{(i)} \tag{17a}$$

and

$$\xi_o^{(i)} = \gamma_o\left(a_o^{(i)} - a_1^{(i)}x_{i-1}\right). \tag{17b}$$

LINE IMPEDANCE

From a practical standpoint, the impedance along a transmission line is a very important quantity because it enters into the calculation of the standing wave ratio and the structural return loss on the line. The impedance is defined as the ratio of the voltage to the current at each point along the line,

$$Z_w(x_p) = \frac{V(x_p)}{I(x_p)} = \frac{-Z(x_p)V(x_p)}{\frac{dV(x_p)}{dx_p}}. \tag{18}$$

Substituting for $V(x_p)$ from (14), we obtain

$$Z_w(x_p) = \frac{Z(x_p)}{\rho(x_p)} = \frac{-2Z(x_p)^2}{\frac{dZ(x_p)}{dx_p} + 2Z(x_p)\gamma_o\tanh\gamma_o\varphi(x_p)}. \tag{19}$$

Since $Z_W(x_p)$ is a function of $\tanh\gamma_o\varphi(x_p)$, an arbitrary degree of accuracy can be maintained by simply retaining more and more terms of the Taylor expansion for $\varphi(x_p)$.

SAMPLE CALCULATIONS

As illustrations of our method, we analyze the exponentially tapered line and the Bessel line. The exponential line is characterized by

$$Z(x) = Z_1 e^{-\alpha x} \quad \text{ohms/wavelength} \tag{20a}$$

and

$$Y(x) = Y_1 e^{\alpha x} \quad \text{mhos/wavelength,} \tag{20b}$$

while, for the Bessel line,

$$Z(x) = Z_1 x \quad \text{ohms/wavelength} \tag{21a}$$

and

$$Y(x) = Y_1/x \quad \text{mhos/wavelength.} \tag{21b}$$

Since these lines are amenable to solution by ordinary techniques, they allow us to test the accuracy of our method. On the pages which follow, we include plots delineating the voltage distribution and impedance over ten wavelengths of exponentially tapered line for the case $Z_1 = Y_1 = 2.0$ and $\alpha = \frac{1}{4}$. In performing our calculations, we set $V_L = 1.0$ and $Z_L = j1.0$. The linear truncation (11) is used with an increment $x_i - x_{i-1} = .0125\lambda$ to specify $\varphi(x)$ along the line. This is followed by use of (16) to determine the solution $V(x)$ and by (19) to calculate $Z_w(x)$. We observe that the approximate result is indistinguishable from the exact solution. In fact, the approximation agrees with the true solution to three significant figures over ten wavelengths of line. In order to display the characteristics of the error, we have plotted it on an expanded scale in Fig. 1b. As might be expected in any initial value calculation, the error tends to increase as the computations progress in the x direction. We note, however, that the error is in the fourth significant figure and that it oscillates about zero. Improvements in the accuracy of the approximation can be achieved by decreasing the increment $x_i - x_{i-1}$ and/or retaining more terms of the Taylor series used in the analytic continuation of $\varphi(x)$.

REFERENCES

[1] Morse, P. M. and Feshbach, H., Methods of Theoretical Physics, Part II, McGraw Hill Book Co., N. Y., 1953.

[2] Wait, J. R., Electromagnetic Waves in Stratified Media, The Macmillan Co., N. Y., 1962.

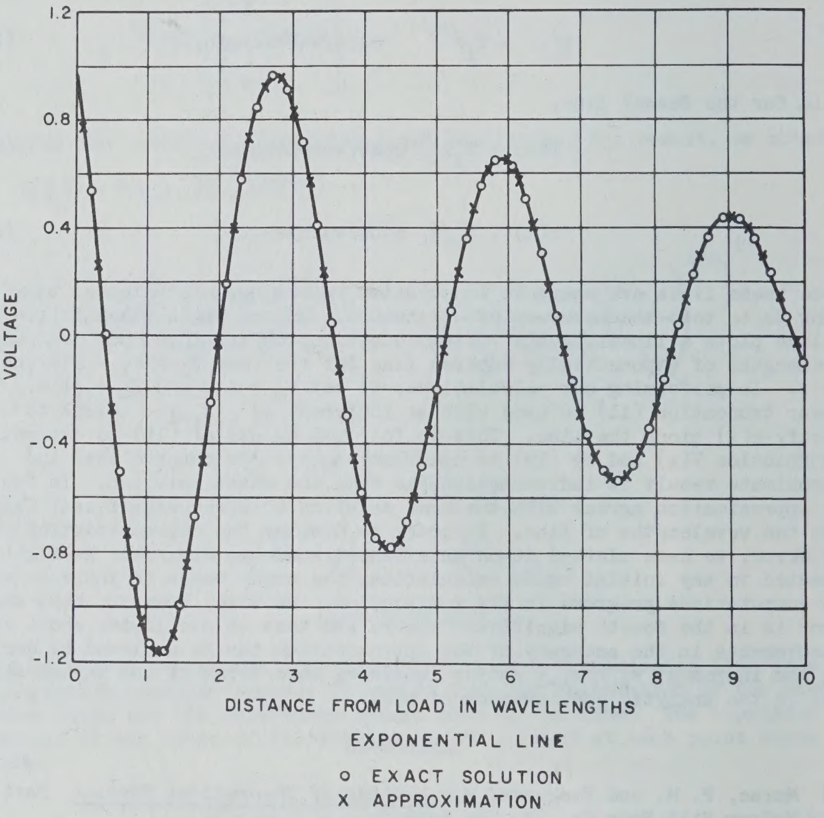


FIG. 1a VOLTAGE DISTRIBUTION $Z_1 = Y_1 = 2.0$

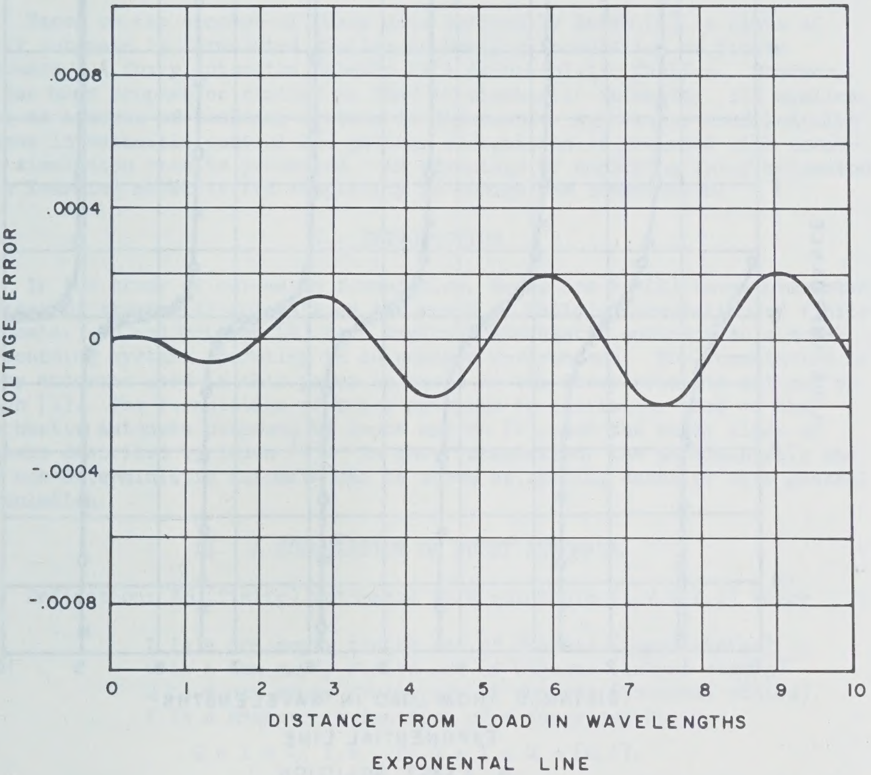


FIG. 1b ERROR CURVE $Z_1 = Y_1 = 2.0$

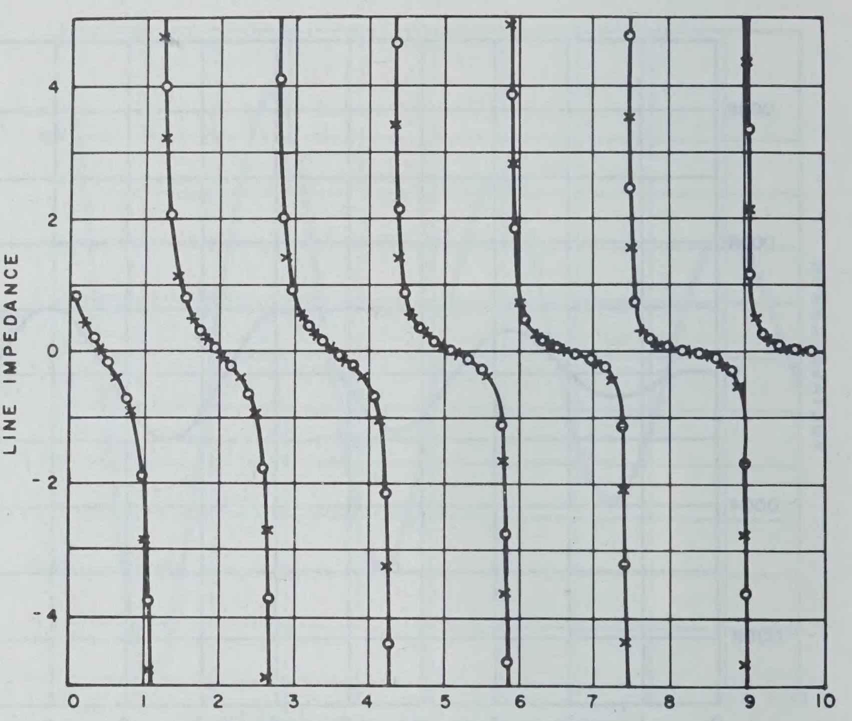


FIG. Ic. LINE IMPEDANCE $Z_1 = Y_1 = 2.0$

A FORMULATION OF FUZZY AUTOMATA AND ITS APPLICATION AS A MODEL OF LEARNING SYSTEMS*

W. G. Wee
Honeywell, Inc.
St. Paul, Minnesota

K. S. Fu
Purdue University
Lafayette, Indiana

ABSTRACT

Based on the concept of fuzzy sets defined by Zadeh [1], a class of fuzzy automata is formulated similar to Mealy's formulation of finite automata. A fuzzy automaton behaves in a deterministic fashion. However, it has many properties similar to that of stochastic automata. Its application as a model of learning systems is discussed. A nonsupervised learning scheme in automatic control and pattern recognition is proposed with computer simulation results presented. An advantage of employing fuzzy automaton as a learning model is its simplicity in design and computation.

I. INTRODUCTION

In the study of automaton formulation, Bruce and Fu [2] have formulated a class of stochastic automata on the basis of Mealy's formulation of finite automata. Fu and McLaren [4] have employed stochastic automata as a model of learning systems operating in an unknown environment. The formulation of fuzzy automata used in this paper is based on the fuzzy relation defined by Zadeh [1]. The formulation of fuzzy automata is similar to that of the stochastic automata proposed by Bruce and Fu [2], and the fuzzy class of systems described by Zadeh [7]. In this formulation, the deterministic and the non-deterministic automata can be shown as special cases of this general formulation.

II. A FORMULATION OF FUZZY AUTOMATA

Definition: An (finite) automaton is a quintuple (I, V, Q, f, g) where

I is a non-empty finite set of objects (input states).
v is a non-empty finite set of objects (output states).
Q is a non-empty finite set of objects (internal states).
f is a membership function of a fuzzy set in

$$Q \times I \times Q; \text{ i.e., } f: Q \times I \times Q \rightarrow [0,1].$$

g is a membership function of a fuzzy set in $V \times I \times Q$; i.e.,

$$g: V \; s \; I \times Q \rightarrow [0,1].$$

It is important to note that I, V, and Q are ordinary sets of finite number of elements, while f and g are membership functions of fuzzy sets of finite number of elements. f may be called the fuzzy transition function and g the fuzzy output function.

Let $I = \{i_1, i_2, \ldots, i_p\}$, $V = \{v_1, v_2, \ldots, v_r\}$, and $Q = \{q_1, q_2, \ldots, q_n\}$, then

$$f_A(q_\alpha, i_j, q_m) = \text{the grade of transition (class A) from state } q_\alpha \text{ or } q(k) = q_\alpha \text{ to } q_m \text{ or } q(k+1) = q_m \text{ when the input is } i_j \text{ or } i(k) = i_j. \quad (1)$$

---

*This work was supported by the National Science Foundation Grant GK-696 and GK-1970.

Therefore,

$$f_A(q_\alpha, i_j, q_m) = f\{q(k)=q_\alpha,\ i(k)=i_j,\ q(k+1)=q_m\}\ .$$

A membership function of unity implies the definite existence of such a transition, and a zero implies no such transition exists. In order to decide the existence of the transition, a pair of thresholds may be introduced as follows: $\alpha > \beta$ where $0 < \alpha,\ \beta < 1$ which leads to a three-level logic as

(i) "x belongs to A" or "true" if $f_A(x) \geq \alpha$
(ii) "x does not belong to A" or "false" if $f_A(x) \leq \beta$
(iii) "x has an indeterminate status relative to "A" or "undetermined" if $\beta < f_A(x) < \alpha$

where A is a fuzzy set defined as the establishment of the transition between states for a particular input; x denotes the 3-type (q_α, i_j, q_m) where $q_\alpha, q_m \in Q$; $i_j \in I$; i.e., $f_A(x) = f(q_\alpha, i_j, q_m)$ as defined in (1).

The function may be dependent or independent of k (the number of steps). If f is independent of k, it is called stationary fuzzy transition function; if f is a function k, it is called nonstationary fuzzy transition function. One will see later that a learning fuzzy automaton must have nonstationary fuzzy transition functions.

For the present time, let f be independent of k, with fuzzy transition matrix T_{i_j} for each $i_j \in I$. The T_{i_j} will be of the following form:

$$\begin{array}{c|cccccc}
 & & & & i_j & & \\
q(k) \diagdown q(k+1) & q_1, & q_2, & \dots, & q_m, & \dots, & q_s \\ \hline
q_1 & & & & \vdots & & \\
\vdots & & & & \vdots & & \\
q_\alpha & \cdots & \cdots & \cdots & f_A(q_\alpha, i_j, q_m) & & \\
\vdots & & & & & & \\
q_n & & & & & & \\ \hline
\end{array} \qquad i_j \in I.$$

The fuzzy transition table will be as

	q(k+1)			v(k)		
q(k) \ i(k)	a	b		a	b	
q_1						
q_2						
⋮	[A]	[B]	. . . .	$[A_o]$	$[B_o]$	. . . .
q_n						

where [A] -- fuzzy matrix formed by $f_A(q_i, a, q_j)$, for all $q_i,\ q_j \in Q$.
$[A_o]$-- fuzzy matrix formed by $g_A(v_j, a, q_i)$, for all $q_i \in Q$, $v_j \in V$.

The input sequence transition matrix for a particular n-input tape sequence is defined by an n-ary fuzzy relation in the product space $T_1 \times T_2 \dots \times T_n$. The fuzzy transition function is as follows: Let $I_j(k)$ be an input sequence of length j, i.e., $i_1(k),\ i_2(k+1), \dots, i_j(k+j-1)$. Then $f_A(q_\alpha, I_p(k), q_s) = f_A(q, i_j, i_o, \dots, i_t, q_s)$ = the grade of transition (class A) from state q_α or $q(k) = q_\alpha$ to q_s or $q(k+p) = q_s$ when the input sequence is $i(k) = i_j,\ i(k+1) = i_o, \dots, i(k+p-1) = i_t$. Therefore,

$f_A(q_\alpha, I_2(k), q_r) = f_A(q_\alpha, i_j, i_o, q_r) = f_A(q_\alpha, i_j, q_m; q_m, i_o, q_r)$, all $q_m \in Q$
$f_A(q_\alpha, I_3(k), q_s) = f_A(q_\alpha, i_j, i_o, i_p, q_s) = f_A(q_\alpha, i_j, i_o, q_r; q_r, i_p, q_s)$, all $q_r \in Q$
$= f_A(q_\alpha, i_j, q_m;\ q_m, i_o q_r; q_r, i_p, q_s)$; all $q_m, q_r \in Q$.

The composition of two fuzzy relations A and B denoted by $B^{o}A$ is defined as a fuzzy relation in X whose membership function is related to those of A and B by $f_{BoA}(x, y) = \mathrm{Sup}_v \mathrm{Min}[f_A(x, v), f_B(v, y)]$. Applying this idea in A^oA we have $f_{AoA}(x, y) = \mathrm{Sup}_v \mathrm{Min}[f_A(x, v), f_A(v, y)]$. This relation serves an intuitive explantion; that is, the pessimistic case is being considered when the minimum f function is selected between $f_A(x, v)$ and $f_A(v, y)$ and the maximal grade of this minimum is being searched through v. Therefore, $f_A(q_\alpha, I_2(k), q_r) = f_A(q_\alpha, i_j, i_o, q_r) = f_A(q_\alpha, i_j, q_m; q_m, i_o, q_r) = \mathrm{Sup}\ q_m\ \mathrm{Min}[f_A(q_\alpha, i_j, q_m), f_A(q_m, i_o, q_r)]$. $m = 1, 2, \ldots, n$; $q_m \in Q$. $f_A(q_\alpha, I_2(k), q_s) = f_A(q_\alpha, i_j\, i_o, q_s) = f_A(q_\alpha, i_j\, i_o, q_r; q_r, i_p, q_s) = f_A(q\alpha, i_j, q_m; q_m, i_o, q_r; q_r, i_p, q_s) = \mathrm{Sup}\ q_r\ \mathrm{Min}\ \{\mathrm{Sup}\ q_m\ \mathrm{Min}[f_A(q_\alpha, i_j, q_m), f_A(q_m, i_o, q_r)], f_A(q_r, i_p, q_s)]\}$. $(r, m = 1, 2, \ldots, n) = \mathrm{Sup}\ q_m, q_r, \mathrm{Min}\{f_A(q_\alpha, i_j, q_m), f_A(q_m, i_o, q_r), f_A(q_r, i_p, q_s)\}$. $m, r = 1, 2, \ldots, n$; $q_m, q_r \in Q$. The last extension can be shown by direct expansion of the terms. In general $f_A(q_\alpha, I_j(k), q_m) = f_A(q_\alpha, I_{j-1}(k) i_j(k+j-1), q_m) = f_A(q_\alpha, i_1(k) i_r(k+1) \ldots i_j(k+j-1), q_m) = \mathrm{Sup}\ q_o, q_p, \ldots, q_s\ \mathrm{Min}[f_A(q_\alpha, i_1, q_o), f_A(q_o, i_2, q_p), \ldots, f_A(q_s, i_j, q_m)]$ where $0, p, \ldots, s = 1, 2, \ldots, n$; $q_o, q_p, \ldots, q_s \in Q$. Example. Let T_{i_1} be as follows:

$q(k)$ \ $q(k+1)$	q_1	q_2	q_3	q_4
q_1	f_{11}	f_{12}	f_{13}	f_{14}
q_2	f_{21}	f_{22}	f_{23}	f_{24}
q_3	f_{31}	f_{32}	f_{33}	f_{34}
q_4	f_{41}	f_{42}	f_{43}	f_{44}

$f_A(q_3, i_1, q_2) = f_{32}$; $f_A(q_3, i_1^2, q_2) = \mathrm{Sup}\ q_x\ \mathrm{Min}[f_A(q_3, i_1, q_x), f_A(q_x, i_1, q_2)]$. $(x = 1, 2, 3, 4) = \mathrm{Sup}\ \{\mathrm{Min}[f_{31}, f_{12}], \mathrm{Min}[f_{32} f_{22}], \mathrm{Min}[f_{33} f_{32}], \mathrm{Min}[f_{34} f_{42}]\}$; $f_A(q_3, i_1^3, q^2) = \mathrm{Sup}\ q_x, q_y\ \mathrm{Min}\ f_A(q_3, i_1, q_x), f_A(q_x, i_1, q_y), f_A(q_y, i_1, q_2), (x, y = 1, 2, 3, 4) = \mathrm{Sup}\{\mathrm{Min}(f_{31} f_{11} f_{12}), \mathrm{Min}(f_{31} f_{12} f_{22}), \mathrm{Min}(f_{31} f_{13} f_{32}), \mathrm{Min}(f_{31} f_{14} f_{42}), \mathrm{Min}(f_{32} f_{21} f_{12}), \mathrm{Min}(f_{32} f_{22} f_{22}), \mathrm{Min}(f_{31} f_{13} f_{32}), \mathrm{Min}(f_{32} f_{24} f_{42}), \mathrm{Min}(f_{33} f_{31} f_{12}), \mathrm{Min}(f_{33} f_{32} f_{22}), \mathrm{Min}(f_{33} f_{33} f_{32}), \mathrm{Min}(f_{33} f_{34} f_{42}), \mathrm{Min}(f_{34} f_{41} f_{12}), \mathrm{Min}(f_{34} f_{42} f_{22}), \mathrm{Min}(f_{34} f_{43} f_{32}), \mathrm{Min}(f_{34} f_{44} f_{42})\}$. Note: $f_A(q_3, i_1^3, q_2) = \mathrm{Sup}\ q_x\ \mathrm{Min}\ f_A(q_3, i_1^2, q_x), f_A(q_x, i_1, q_2) (x = 1, 2, 3, 4) = \mathrm{Sup}\{\mathrm{Min}[f(q_3, i_1^2, q_1), f_{12}], \mathrm{Min}[f(q_3, i_1^2, q_2), f_{22}], \mathrm{Min}[f(q_3, i_1^2, q_3), f_{32}], \mathrm{Min}[f(q_3, i_1^2, q_4), f_{42}]\}$. This last relation serves as an iteration scheme for a particular sequence of input tape.

III. SPECIAL CASES OF FUZZY AUTOMATA

1. Deterministic automata. The set "A" is no longer a fuzzy set; it is an ordinary set where f takes only two values, 0 and 1. Therefore the entires for each row of matrices [A], [B], [C], ..., $[A_o]$, $[B_o]$, ... will have only one "1" and the rest "0". So the skeleton matrix [D] of the deterministic automaton will be [D] = [A] + [B] + [C] + ... + [H] which is a reduction similar to that in the stochastic automata formulated by Bruce and Fu [2]. For a two-step transition chain, $f_A(q_\alpha, i_j i_o, q_r)$ will consist of finding the maximum of the minimum of pairs of value of 4-types as (0, 0), (0, 1), (1, 0), (1, 1). Certainly, in order to get a "1" for $f_A(q_\alpha, i_j i_o, q_r)$, if the path of length 2 exists for input sequence $I_2 = i_j i_o$, it must have a pair (1, 1). The total number of paths of length 2 from state q_α to state q_r will be equal to

$$\gamma_{\alpha r} = \sum_{i_j i_o} f_A(q_\alpha, i_j i_o, q_r) .$$

Similarly, for a path of length 3 to exist from q_α to q_r the same operation holds as

$$\sum_{i_j, i_o, i_p} f_A(q_\alpha, i_j i_o i_p, q_r) = \sum_{i_j, i_o, i_p} \underset{x,y=1,2,\ldots,n}{\text{Sup } q_x\, q_y} \text{ Min}\left[f_A(q_\alpha, i_j, q_x),\ f_A(q_x, i_o, q_y),\ f_A(q_y, i_p, q_r)\right].$$

It consists of terms as (1,1,1), (1,0,1),...,(0,0,0); of course, only (1,1,1) defines the existence of the path.

2. Non-deterministic automata. The following definition of a non-deterministic automaton has been used by Ginsburg [3].

A non-deterministic automaton is a quintuple $A = (K, \Sigma, \delta, S_o, F)$, where

(i) δ is a function from $K\times\Sigma$ into 2^K (this includes the empty set $\emptyset$)

(ii) S_o and F are subsets of K, with S_o non-empty,

with K a non-empty finite set of objects (internal states) and Σ a non-empty finite set of objects (input states).

To fit this model from the fuzzy automata formulation, the set "A" is no longer a fuzzy set; it is an ordinary set where f takes only two values 0 and 1. The entries for each row of the matrices [A], [B], [C],... have either "1" or "0". No restriction has been made to the number of 1's in each row of the matrices. If all the entries of a particular row are zero, then the transition maps the state into the empty set $\emptyset$. If a particular row has more than one "1" then the transition may map the succeeding state into anyone of the possible states $(q_i, q_j, \ldots)$. For example, if T_{i_1} is given as

$$T_{i_1} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix},$$ then the mapping is as

$q(k+1)$ $q(k)$	$i(k)$ i_1
q_1	$\{q_1, q_2\}$
q_2	$\emptyset$
q_3	$\{q_1, q_2, q_3\}$

3. Normalized fuzzy automata and its relation to stochastic automata. One important observation is that the membership function "f" (from its definition) does not reveal too much of the nature of the function except satisfying the two conditions given below:

(i) f is a "grade of membership" function defined for each $x \in X$.

(ii) f is in the close interval [0,1].

Thus it is felt that "f" function is so general that extra constraints may be incorporated as long as the two conditions (i) and (ii) are fulfilled. A very special restriction will be the row sum of all transition matrices equal to unity similar to that in stochastic automata. In other words, [A], [B], [C], --- have exactly the same structure as that of stochastic matrices. This type of automata can be called Normalized Fuzzy Automata. There are many properties that can be explored for fuzzy automata similar to those for deterministic automata and stochastic automata [6].

IV. FUZZY AUTOMATA AS MODELS OF LEARNING SYSTEMS

A basic learning system is given in Figure 1. The proposed model represents a nonsupervised learning system if a proper performance evaluator can be selected. The decision maker operates deterministically. The learning section is a fuzzy automaton. The performance evaluator serves as an unreliable "teacher" [5] who tries to teach the "student" (the learning section and the decision maker) to make right decisions. The performance of the system monotonically approaches to the desired solution if the path of learning is a right one. Once it has been proved wrong, the system steps back a number of steps to relearn the new environment. The back-stepping is described by increasing the estimated grade of membership of a particular class which will be explained in detail in the following paragraphs.

The decision executed by the decision maker is deterministic. Since "on-line" operation is required, the decision will be based on the maximum grade of membership; i.e., decide $x \in \omega_i$ if $f_{\omega_i}(x) = \operatorname{Max}_j f_{\omega_j}(x)$ with $\omega_i, \omega_j \in \Omega$; $i, j=1,2,\ldots,N$ where $\Omega = \{\omega_i : i=1,2,\ldots,N\}$ denotes the set of pattern classes.

If the decision maker is allowed to stay undecided and defer its decision, especially at the beginning of the first few learning steps, then we have to decide $x \in \omega_i$ if $f_{\omega_i}(x) = \operatorname{Max}_j f_{\omega_j}(x) \geq C$; undecided if $f_{\omega_i}(x) = \operatorname{Max}_j f_{\omega_j}(x) < C$; $\omega_i, \omega_j \in \Omega$; $i, j=1,2,\ldots,N$ where $0 < C \leq 1$. Under this condition, the decision maker is allowed to use a pure random strategy for making decisions until it can make a decision with a certain degree of confidence.

Once the learning section proceeds into an erroneous situation due to unreliable information obtained from the performance evaluator, it will continue making erroneous decisions until the teacher corrects its own error; then, in turn, the teacher will force the learning section to relearn the new situations.

The learning section is represented by a fuzzy automaton. For an n-state fuzzy automaton, let $f_j(k)$ be the grade of membership for the automaton to be at state q_j in step k. The entries of the fuzzy transition matrix are denoted by $f_{ij}^{\alpha}(k) = f\{q(k) = q_i,\ i(k) = i_{\alpha},\ q(k+1) = q_j\}$. Then for the input $i(k) = i_{\alpha}$, $\hat{f}_j(k+1) = \operatorname{Sup}_m \operatorname{Min}\{f_m(k), f_{mj}^{\alpha}(k)\}$, $m=1,2,\ldots,n$, $= \operatorname{Sup}\{\operatorname{Min}[f_m(k), f_{mj}^{\alpha}(k)];\ \operatorname{Min}[f_j(k), f_{jj}^{\alpha}(k)]\}$, $m=1,2,\ldots,n$, $m \neq j$.

It is obvious that in order to have learning behavior, the fuzzy transition matrix must be nonstationary. In order to simplify the problem, let

$$f_{mj}^{\alpha}(k) = c_j^{\alpha}(k) \quad \text{for all } m \neq j \quad \text{and} \quad f_{jj}^{\alpha}(k) = \alpha_j f_{jj}^{\alpha}(k-1) + (1-\alpha_j)\lambda_j$$

with

$$f_{ij}^{\alpha}(k) = c_j^{\alpha}(k) = \begin{cases} 0 \text{ if } k \text{ is odd} \\ f_{jj}(k-1) \text{ if } k \text{ is even} \end{cases} \qquad \begin{array}{l} \text{where } 0 < \alpha_j < 1,\ 0 < \lambda_j \leq 1 \\ j = 1,2,\ldots,n \end{array}$$

Furthermore, let

$$f_j(k+1) = \begin{cases} f_{jj}^{\alpha}(k) + a_j \text{ if } f_{jj}^{\alpha}(k) - \hat{f}_j(k+1) \geq b_j & (2) \\ \qquad 0 < a_j,\ b_j < 1 & \\ \hat{f}_j(k+1) = \operatorname{Sup}_m \operatorname{Min}\{f_m(k), f_{mj}^{\alpha}(k)\}, \text{ otherwise} & (3) \end{cases}$$

where a_j may take the form of $a_j = d_j(f_{jj}^{\alpha}(k) - \hat{f}_j(k+1))$ or $a_j = d_j b_j$, $0 < d_j \leq 1$; b_j is a bounded sequence such that $b_j \to 0$ as $j \to \infty$. Therefore, $a_j \to 0$ as $j \to \infty$. With these assumptions, $f_j(k+1)$ is always between 0 and 1.

Equation (3) denotes the regular learning behavior which has monotonic property [4]; with a perfect teacher, the fuzzy automaton converges to λ_j, $j=1,2,\ldots,n$. Equation (2) is the relearning procedure; i.e., the automaton is in an erroneous situation and must relearn the new situations by stepping backward a number of steps. The backward stepping is also described by equation (2). The number of backward steps required is controlled by d_j and b_j at the j-th step of learning.

The fuzzy automaton starts with no a priori information $f_j(0) = 0$ or 1 for all j, or with some a priori informations $f_j(0) = \hat{\lambda}_j$, for all j, $0 < \hat{\lambda}_j \leq 1$. With a perfect teacher, λ_j, $j=1,\ldots,n$ are known a priori. The given algorithm converges monotonically to the desired solution; i.e.,

$$[f_1(k), f_2(k), \ldots, f_n(k)] \to [\lambda_1, \lambda_2, \ldots, \lambda_n] \text{ as } k \to \infty.$$

The convergence property can be verified, as $f_j(k+1)$ depends on $f_{ij}(k)$ and $c_j(k)$, while $f_{jj}(k) = \alpha_j f_{jj}(k-1) + (1-\alpha_j)\lambda_j = \alpha_j^k f_{jj}(0) + (1-\alpha_j^k)\lambda_j$, $0 < \alpha_j < 1$ and $\lim_{k \to \infty} f_{jj}(k) = \lambda_j$. Therefore, $f_j(k) \to \lambda_j$ as $k \to \infty$.

The typical learning curves with a perfect teacher are given in Fig. 2. The rate of learning depends on α_j. The typical learning curves with an unreliable teacher are shown in Fig. 3.

VI. APPLICATIONS AND SIMULATION RESULTS

The learning model proposed above has been applied to engineering problems. The emphasis will be placed upon its applications to pattern classification and control systems. The performance evaluator $M(Z,k)$ must be such that $0 < M(Z,k) \leq T < \infty$, $k = 1,2,\ldots$ and $\lim_{k\to\infty} M(Z,k) = M > 0$. The goal of the system is to maximize or minimize $\lim_{k\to\infty} M(Z,k)$. $Z(k)$ is usually termed as the instantaneous performance evaluation at k-th step of learning. Thus $M(Z,k)$ is an estimator of $Z(k)$ at k-th step of learning. As an example,

$$M(Z,k) = \frac{1}{K} \sum_{i=1}^{k} Z(i) = \frac{k-1}{k} \left(\frac{1}{k-1} \sum_{i=1}^{k-1} Z(i)\right) + \frac{Z(k)}{k} = \frac{k-1}{k} M(Z,k-1) + \frac{Z(k)}{k}$$

$$k = 1,2,\ldots$$

In this particular case, $M(Z,k)$ is a sample average of $Z(k)$ estimated recursively at each step of learning.

1. Application to pattern classification. A pattern recognition system with non-supervised learning is given in Fig. 4. The role of input and output can be briefly explained as follows: During each time interval, the pattern classifier receives a new sample X' from the unknown environment. After X' is processed through the receptor, the output is fed to both the decision maker for classification and the performance evaluator for evaluation. The performance criterion for the system must be selected so that its maximization or minimization reflects the clustering properties of the pattern classes; i.e., the unknown environment. Therefore, due to the natural distribution of the samples, performance criterion can be incorporated into the system to serve as a "teacher" of the learning pattern recognizer.

The non-supervised learning model as applied to the problems of pattern classification is formulated as follows: It is assumed that the classifier (the decision maker) has at its disposal sets of discriminant functions characterized by sets of parameters. With a proper specification of the performance evaluation and without any external supervision, the system adapts itself to the best solution. The best solution denotes the set of discriminant functions that gives the minimum misrecognition among the sets of discriminant functions for the given set of training samples. The criterion to be used in this particular case is based upon the sample averages and the average deviations from the sample averages of training patterns generated by each set of discriminant functions. The best set of discriminant functions must give the maximum total distance between its sample averages and the minimum total "sample deviation" (average of the squared deviation from the sample averages).

Let $\Omega = \{\omega_i;\ i=1,2,\ldots,R\}$ be the set of pattern classes, and $X_1, X_2, \ldots$ be the sequence of incoming samples from the unknown environment in E^d. N sets of discriminant functions are given a priori. The decision maker is as shown in Fig. 5. Then,

$$K_k^{\alpha} = \left| w_1 \left[\sum_{i=1}^{R} S_{ki}^{\alpha} \right] - w_2 \left[\sum_{i>j} (M_{ki}^{\alpha} - M_{kj}^{\alpha})^T (M_{ki}^{\alpha} - M_{kj}^{\alpha}) \right] \right|$$

$$\alpha = 1,2,\ldots,N$$

where K_k^{α} -- the performance evaluation for α-th set of discriminant functions at the k-th step of learning. S_{ki}^{α} -- the "sample deviation" for the i-th class of the α-th set of discriminant functions at the k-th step of learning. M_{ki}^{α} -- the sample average for the i-th class of the α-th set of discriminant functions at the k-th step of learning. $0 \leq w_1, w_2 < \infty$.

The values of the K_k^α, S_{ki}^α, and M_{ki}^α can be estimated recursively as follows: For α-th set of discriminant functions, if $X_{k+1} \in \omega_i$, $i=1,2,\ldots,R$ then

$$M_{k+1,i} = \frac{N_{k,i}}{N_{k,i}+1} M_{k,i} + \frac{X_{k+1}}{N_{k,i}+1}$$

$$S_{k+1,i} = \frac{1}{N_{k,i}+1} \sum_{j=1}^{N_{k,i}+1} (X_j - M_{k+1,i})^T (X_j - M_{k+1,i}).$$

After expansion and simplification, we have

$$S_{k+1,i} = \frac{N_{k,i}}{N_{k,i}+1} S_{k,i} + \frac{N_{k,i}}{(N_{k,i}+1)^2} (X_{k+1} - M_{k,i})^T (X_{k+1} - M_{k,i}).$$

Also, $M_{k+1,j} = M_{k,j}$ for $j \neq l$; $S_{k+1,j} = S_{k,j}$ for $j \neq i$. $N_{k,i}$ denotes the number of samples belonging to ω_i up to k-th step of learning. The minimum K_k^α, $\alpha = 1,2,\ldots,N$ serves to indicate the best solution among the N sets of discriminant functions.

Specific computer simulations are conducted for (1) two equal numbers of training characters from A and B using one hyperplane, and (2) three equal numbers of training characters from A, B and C using three hyperplanes.

The estimation of the N membership functions, $\hat{\lambda}_j(k)$, at k-th step of learning is as follows:

$$\hat{\lambda}_j(k) = 1 - \frac{K_k^j}{C}, \quad \begin{array}{l} j = 1,2,\ldots,N \\ k = 1,2,\ldots \end{array}$$

$$C \geq \max_{k,j} K^j$$

with

$$\lim_{k\to\infty} \hat{\lambda}_j(k) = \lambda_j, \quad 0 \leq \lambda_j \leq 1.$$

The description of the feature extraction is given in Fig. 6. The patterns used in the computer simulations are the three English capital characters "A", "B", and "C". Each character is handwritten within a two-inch square frame. For each character, four features, $x_1, x_2, x_3 \cdot x_4$ are extracted. The features extracted correspond to the number of distinct intersections of the sample character with lines a_1, a_2, a_3, a_4 as indicated in Fig. 6.

The details of the computer flow diagram are shown in Fig. 7. Ten sets of hyperplanes are used (predetermined) in both examples with 60 samples from each class. The incoming samples are introduced to the system in a random fashion. The results of the learning curves, $f_j(k)$ vs k for each example, are given in Figs. 8 and 9.

2. Application to control system. The learning controller proposed here is similar to that in [5]. It may be briefly explained as follows: A time-discrete plant may be described as $x(k+1) = \varnothing_{k+1}[x(k), u(k+1)]$ where $x(k), x(k+1) \in \Omega_u \triangleq \{x_i, i=1,2,\ldots,R;\ R < \infty\}$ and $u(k), \in \Omega_u \triangleq \{u_i, i=1,2,\ldots,P;\ P < \infty\}$. $x(k)$ is the observed response of the plant at the (k+1)-th instant when the control action $u(k+1)$ is applied. It is assumed that $\varnothing_k$ is not known for $k=1,2,\ldots$ The instantaneous performance evaluation of a control action $u(k)$ is given by $Z(k+1) = g(x(k), u(k+1), x(k+1))$ with $0 < Z(k) < T < \infty$ for $k=1,2,3,\ldots$ The goal of the control is to minimize $\hat{M}_{k+1}(Z|u(k), x(k), u(k+1))$, the sample average of Z. The sample average for each control policy is estimated as follows: Let $u(k+1) = u_\alpha$ be applied after observing $u(k) = u_j$ and $x(k) = x_i$. Then

$$\hat{M}_{k+1}[X|u_j, x_i, u_\alpha] = \frac{N}{N+1} \hat{M}_k[Z|u_j, x_i, u_\alpha] + \frac{1}{N+1} Z(k+1)$$

$$\hat{M}_{k+1}[Z|u_j, x_i, u_h] = \hat{M}_k[Z|u_j, x_i, u_h], \quad h = 1,2,\ldots,P, \quad h \neq \alpha$$

where $N = N(j,i,\alpha)$ denotes the number of occurrences of $u(k) = u_j$, $x(k) = x_i$,

$u(k+1) = u_\alpha$.

$$\hat{f}_{k+1}[u_\alpha / u_j, x_i] = 1 - \frac{\hat{M}_{k+1}[Z \mid u_j, x_i, u_\alpha]}{T} . \tag{4}$$

Equation (4) enables the system to associate the control action with the maximum grade of membership with the minimum sample average of Z. Fig. 10 illustrates the block diagram of the learning controller.

VII. CONCLUSIONS

A class of fuzzy automata is formulated. Its application as a model of learning systems is proposed. A non-supervised learning algorithm for a fuzzy automaton is presented, together with its application to pattern recognition and automatic control problems. Computer simulations of character recognition indicate satisfactory results.

VIII. REFERENCES

1. Zadeh, L. A., "Fuzzy Sets", Information and Control, Vol. 8, No. 3, June, 1965, pp. 338-353.

2. Bruce, G. D. and Fu, K. S., "A Model for Finite-State Probabilistic Systems", Proceedings of the Allerton Conference on Circuit and System Theory, University of Michigan Press, 1963.

3. Ginsburg, S., An Introduction to Mathematical Machine Theory, Addison-Wesley Publishing Company, Inc., 1962.

4. Fu, K. S. and McLaren, R. W., "An Application of Stochastic Automata to the Synthesis of Learning Systems", Tech. Rept., TR-EE65-17, School of Electrical Engineering, Purdue University, April, 1965.

5. Nikolic, Z. J., and Fu, K. S., "An Algorithm for 'Learning Without External Supervision' and Its Applications to Learning Control Systems", IEEE Transactions on Automatic Control, July, 1966.

6. Wee, W. G., "On Generalizations of Adaptive Algorithm and Application of the Fuzzy Sets Concept to Pattern Classification", Ph.D Thesis, Purdue University, Lafayette, Indiana, June, 1967.

7. Zadeh, L. A., "Fuzzy Sets and Systems", Proceedings of the Symposium on System Theory, Polytechnic Institute of Brooklyn, April 20, 21, 22, 1965.

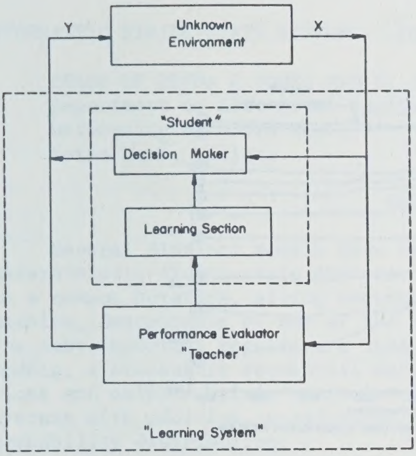

Y
Unknown Environment
X
"Student"
Decision Maker
Learning Section
Performance Evaluator "Teacher"
"Learning System"

FIGURE 1. A BASIC LEARNING MODEL

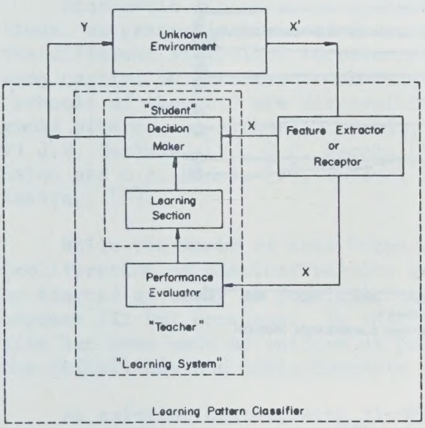

Y
Unknown Environment
X'
"Student"
Decision Maker
X
Feature Extractor or Receptor
Learning Section
Performance Evaluator
X
"Teacher"
"Learning System"
Learning Pattern Classifier

FIGURE 4. BLOCK DIAGRAM OF A LEARNING PATTERN CLASSIFIER

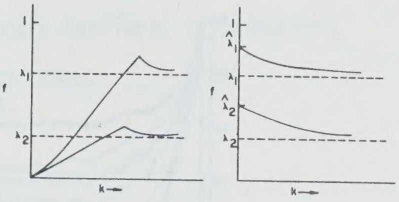


FIGURE 2. LEARNING CURVES WITH PERFECT TEACHER

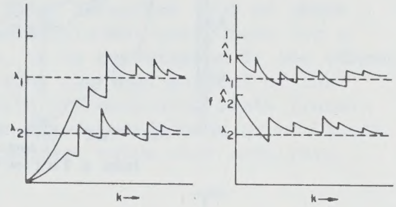
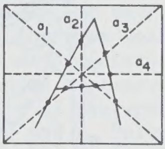


FIGURE 3. LEARNING CURVES WITH UNRELIABLE TEACHER



Sample Feature: $[x_1, x_2, x_3, x_4] = [3232]$

FIGURE 5. DESCRIPTION OF FEATURE EXTRACTION

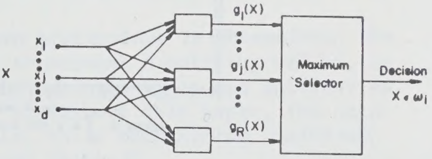

X
x_1
x_j
x_d
$g_1(X)$
$g_j(X)$
$g_R(X)$
Maximum Selector
Decision
$X \in \omega_i$

FIGURE 6. A MULTICLASS PATTERN CATEGORIZER

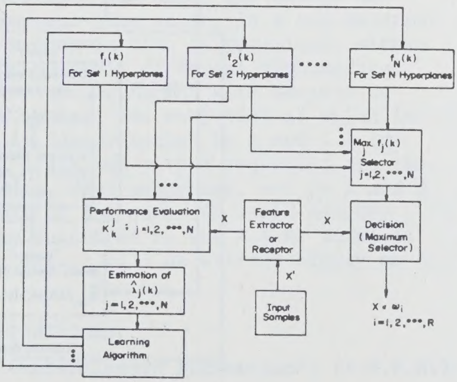

$f_1(k)$ For Set 1 Hyperplanes
$f_2(k)$ For Set 2 Hyperplanes
$f_N(k)$ For Set N Hyperplanes
Max $f_j(k)$ j Selector $j=1,2,\cdots,N$
Performance Evaluation K_k^j; $j=1,2,\cdots,N$
X
Feature Extractor or Receptor
X
Decision (Maximum Selector)
Estimation of $\hat{\lambda}_j(k)$ $j=1,2,\cdots,N$
X'
Input Samples
$X \in \omega_i$ $i=1,2,\cdots,R$
Learning Algorithm

FIGURE 7. COMPUTER FLOW DIAGRAM FOR A LEARNING CHARACTER CLASSIFIER

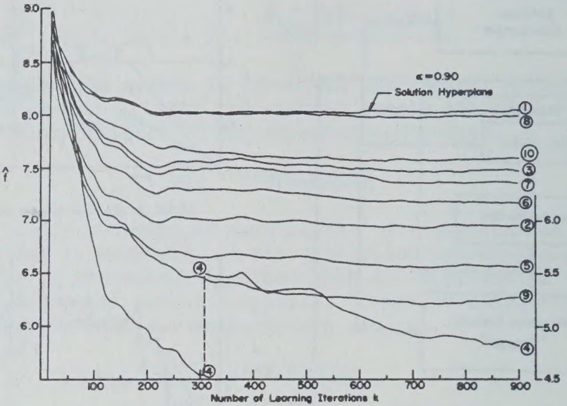


FIGURE 8. $\hat{f}$ vs. k FOR TWO-CLASS PATTERN CLASSIFICATION PROBLEM

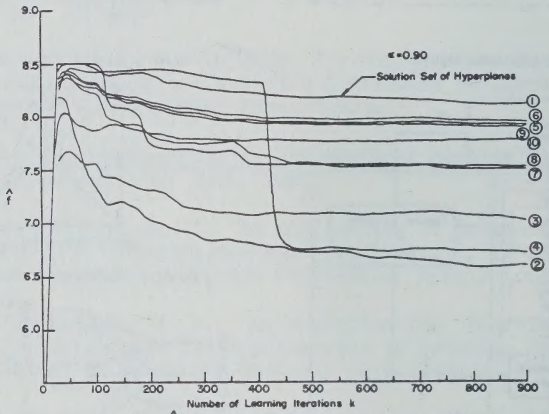


FIGURE 9. $\hat{f}$ vs. k FOR 3-CLASS PATTERN CLASSIFICATION PROBLEM

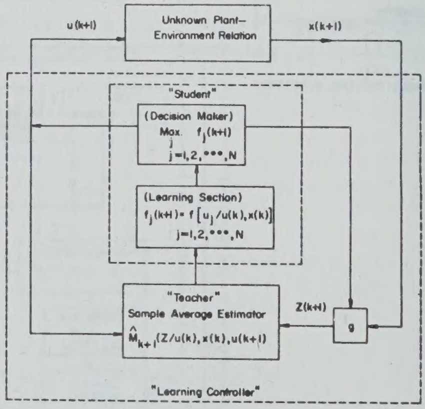

u(k+1)
Unknown Plant–
Environment Relation
x(k+1)
"Student"
(Decision Maker)
Max. $f_j(k+1)$
j
j=1,2,•••,N
(Learning Section)
$f_j(k+1)= f\left[u_j/u(k),x(k)\right]$
j=1,2,•••,N
"Teacher"
Sample Average Estimator
$\hat{M}_{k+1}(Z/u(k),x(k),u(k+1)$
Z(k+1)
g
"Learning Controller"

FIGURE 10. BLOCK DIAGRAM OF THE PROPOSED LEARNING CONTROLLER

STOCHASTIC FINITE-STATE SYSTEMS : FOUR MODELS AND THEIR RELATIONSHIPS

CELSO DE RENNA E SOUZA AND R. JEFFREY LEAKE.
Department of Electrical Engineering,
University of Notre Dame,
Notre Dame, Indiana.

ABSTRACT

Several distinct models have been proposed and used to study non-deterministic finite-state systems. The paper describes four of these in a common notation, giving necessary and sufficient conditions for a machine, describable by one of the models, to be describable by the others. The most important results are those relating the most general of the models, a stochastic sequential machine with probabilistic state transitions and outputs, given by its stochastic matrix, to linear or nonlinear systems with additive, statistically independent noise with arbitrary probability distribution.

I. INTRODUCTION

Stochastic finite-state systems appeared naturally, at different times, as generalizations of deterministic models. For this reason, the different viewpoints represented by the various deterministic models were carried on into the probabilistic realm, and today at least three "schools of thought" are discernible; those composed by the works of, among others: 1) Michael O. Rabin, [1], A. Paz, [2] and A. Salomaa, [3] ; 2) J.W. Carlyle, [4], G.C. Bacon, [5], and G.H. Ott, [6] ; and 3) J.H. Eaton and L.A. Zadeh, [7], Dorato, Drenick and Shaw, [8], and R.L. Kashyap, [9].

While the works of each "school" are meritorious in themselves, the proliferation of distinct results seems to warrant a unifying effort, to be started as early as possible, before the hiatus widen and unification becomes all but hopeless. To this end, throughout this paper, the notation has been made as uniform as possible, while maintaining unaltered the definitions and basic concepts of each "school".

An existing system, with finitely many distinguishable states, inputs and outputs is assumed. Inputs are fed, the state may change and outputs are produced at points on a discrete time scale, in a non-deterministic fashion. The evolution of the system in time is described, within the framework of probability theory, by possibly different mathematical models. We assume the system to be first-order Markov with respect to states and inputs. We will say that a system, the evolution of which is given probabilistically by a model A, is "describable" by a model B iff the probabilities of the corresponding state and output sequences are the same in both models for all corresponding input sequences, and in A and B the number of inputs, states and outputs is the same. To avoid lengthy statements, phrases such as : "A system described by the S.S.M. model is describable by the A.N.M. model iff .. etc.," will be written simply as : "An S.S.M. is an A.N.M. iff etc."

2. FOUR DEFINITIONS

Definition 1. A (initialized) "Moore Probabilistic Automation", (I.M.P.A.) [1], [2], is a system

$$\text{I.M.P.A.} = \langle X,M,S,s_1,F \rangle$$

where

X : is a finite input alphabet, $X = \{x_1,\ldots,x_n\}$;

S : is a finite set of states, $S = \{s_1,\ldots,s_r\}$,

M : is a function, $M : S \times X \to [0,1]^r$, such that

$\forall (s,x) \in S \times X$, $M(s,x) = [p_1(s,x),\ldots,p_r(s,x)]$,

$\forall i$, $p_i(s,x) \geq 0$ and

$$\sum_{i=1}^{r} p_i(s,x) = 1 \; ;$$

$s_1 \in S$: is a given initial state,

$F \subset S$: is the set of "final" states.

The indication of the output by a set of final states is tantamount to giving an output function $g: S \to Y$, for $Y = [0,1]$, such that

$F = \{s : s \in S.\ g(s) = 1\}$.

A more general "Moore" P.A. would have an arbitrary, finite, output alphabet $Y = \{y_1,\ldots.y_m\}$, while still an output function g on S, $g : S \to Y$.

Definition 2.

A "Finite State Channel",(F.S.C.),[10], is a system

$$\text{F.S.C.} = \langle X,Y,S,M_1,\ldots, M_n,g \rangle$$

where X,S, are as for the I.M.P.A.,

Y : is a finite output alphabet, $Y = \{y_1,\ldots,y_m\}$

$M_1,\ldots,M_n$: is a collection of $r \times r$ stochastic matrices the rows for which sum to 1), one for each input symbol,

g : is the output function **g** : $S \times X \to Y$.

The F.S.C. is the probabilistic version of a Mealy machine with deterministic output. Its operation (in continuous time) is the following :

1) The machine is in some state $s_i \in S$.
2) An input $x_j \in X$ is fed.
3) The machine transits to some $s_k \in S$, according to the probabilities in M_j.
4) An output $g(s_k, x_j)$ is produced.

Note that the output is a function of the new state, and not the former.

As our results to date indicate, the most general non-deterministic machine model studied is the next one, first proposed by Carlyle as a generalization of a finite-state model of C. Shannon, for a communication channel :

Definition 3.

A "Stochastic Sequential Machine", (S.S.M.), [4], [6], is a system

$$\text{S.S.M.} = \langle X,Y,S,M, \rangle \;,$$

where X,Y, and S are as before,and M is an $rn \times rm$ stochastic matrix,

$$M = \begin{vmatrix} M(y_1/x_1) & \ldots\ldots\ldots & M(y_m/x_1) \\ & \vdots & \\ M(y_1/x_n) & \ldots\ldots\ldots & M(y_m/x_n) \end{vmatrix}$$

the rows of which are labelled by the elements of S x X and the columns by those of S x Y, where each of the submatrices $M(y_i/x_j)$ is r x r, corresponding to the input x_j and output y_i, and the typical entry of which is

$$\text{Pr. } [(s_u,y_j) \mathbin{/} (s_k,x_i)] \text{ .}$$

Note that both the state transitions and the output are non-deterministic. The state transition matrices are obtained by

$$M_i = \sum_{j=1}^{m} M(y_j/x_i) \quad , \quad i = 1,2,\ldots,n,$$

that is, by adding all the submatrices in a given row.

A different type of model, closer to the physical realisations of machines, that is, in the sequential circuit class, may be put forth. Let us suppose that a deterministic, noninitialized, Mealy machine, linear or not, is given, already "state assigned" :

$$\left\langle \underline{S}, \underline{X}, \underline{Y}, \delta, \lambda \right\rangle$$

where $\underline{X}$, $\underline{Y}$ and $\underline{S}$ are as before, but now are vector spaces over some finite field, and δ and λ are the next-state and output functions (not necessarily homomorphisms),

$$\delta : \underline{S} \times \underline{X} \rightarrow \underline{S}$$

$$\lambda : \underline{S} \times \underline{X} \rightarrow \underline{Y} \text{ ,}$$

which are made then non-deterministic by the insertion of "noise". By analogy with well known continuous and discrete time, continuous state models, the noise will be considered statistically independent and additive, the actual next-states and outputs being given by

$$\delta_N(s,x) = \delta(s,x) + n_\delta$$

$$\lambda_N(s,x) = \lambda(s,x) + n_\lambda \text{ ,}$$

for $s \in \underline{S}$, $x \in \underline{X}$, $n_\delta \in \underline{S}$, $n_\lambda \in \underline{S}$, where n_δ and n_λ are "noise vectors" for which probability frequency functions (distributions) are given.
For instance, letting

$$P = \{p_1, \ldots\ldots p_r\}$$

such that $p_i \in [0,1]$, $\sum_i p_i = 1$, then a 1 : 1 function P_δ is given

$$P_\delta : \underline{S} \rightarrow P$$

and from $\sum_i p_i = 1$,

$$\sum_{\underline{S}} P_\delta(n_\delta) = 1 \text{ ,}$$

and similarly for n_λ.

Definition 4.

The above model will be named "Additive Noise Machine", (A.N.M.).

Clearly, given an A.N.M., the corresponding S.S.M. is easily determined; that is, the entries of M are directly calculated from the knowledge of P_δ and P_λ, without changing the state, input and output spaces. The same holds for passing from a noninitialized M.P.A. to a F.S.C. and to a S.S.M. without relabelling inputs, outputs and states. Pictorially, the transitions

$$\text{M.P.A.} \Rightarrow \text{F.S.C.} \Rightarrow \text{S.S.M.} \Leftarrow \text{A.N.M.}$$

are trivial. The converse problems are not so simple,

$$\text{M.P.A.} \Leftarrow \text{F.S.C.} \Leftarrow \text{S.S.M.} \Rightarrow \text{A.N.M.}$$

and are those treated in this paper.

3. RELATIONSHIPS AMONG THE MODELS

Theorem 1.

An S.S.M. is an F.S.C., with the same input, output and state sets, iff each row of submatrices

$$M(y_1/x_i)\ , \quad \ldots\ldots,\ M(y_m/x_i), \qquad i = 1,2,\ldots,r.$$

has at least r(m-1) all-zero columns.

Proof :

The typical entry of M can be written

$$\Pr.\ [(s_u,y_j)\ /\ (s_k,x_i)] = \Pr.\ [s_u/(s_k,x_i)]\cdot\Pr.[y_j/(s_u,s_k,x_i)]\ ;$$

but, for a F.S.C.,

$$\Pr.\ [y_j/(s_u,s_k,x_i)] = \begin{cases} 1 & \text{if } g(s_u,x_i) = y_j \\ 0 & \text{otherwise} \end{cases}$$

(the distinction between a deterministic output and that happening with probability 1 is not being made : if wished, proper statements could be introduced in Definitions 1., 2., and 4.)

Hence

$$\Pr.\ [(s_u,y_j)/(s_k,x_i)] = \begin{cases} \Pr.\ [s_u/(s_k,x_i)] & \text{if } g(s_u,x_i) = y_j \\ 0 & \text{otherwise} \end{cases}$$

Since, for each x_i, there are at most r possible outputs $g(s_u,x_i)$, they will correspond to the, possibly, non-zero columns in the row of submatrices. All other r(m-1) columns must be all zero.

The converse is immediate. Q.E.D.

Theorem 2.

An S.S.M. is an M.P.A. with same input, output and state sets, iff M itself has at least r(m-1) all-zero columns.

Proof :

In this case

$$\Pr.\ [y_j/(s_u,s_k,x_i)] = \Pr.\ [y_j/s_u] = \begin{cases} 1 & \text{if } g(s_u) = y_j \\ 0 & \text{otherwise} \end{cases}$$

Hence

$$\text{Pr.}\ [(s_u,y_j)\ /\ (s_k,x_i)] = \begin{cases} \text{Pr.}[\ s_u/(s_k,x_i)] & \text{if } g(s_u) = y_j \\ 0 & \text{otherwise} \end{cases}$$

It follows that at most r columns can be nonzero. The converse is immediate.

Q.E.D.

For next theorems, those dealing with the transition S.S.M. ⇒ A.N.M., it is assumed that the set $\mathcal{P}$ is composed of distinct numbers, that is, $p_i \neq p_j$ if $i \neq j$ for $p_i, p_j \in \mathcal{P}$; also, the binary A.N.M. is assumed to be completely specified, that is,

$$r = 2^R$$

$$m = 2^M$$

$$n = 2^N$$

for some integers R,M,N. Only the state transitions are considered, the treatment for the noisy output being a natural extension; out attention will focus on the matrices M_i. The phrase "with respect to state transitions" is to be understood as a premise on all the following theorems.

Theorem 3.

An S.S.M. is an A.N.M. with distinct p_k only if :

i) the rows $\underline{m}_i^l$ of the matrices M_i, for $l = 1,2,\ldots,r$ and all i, are permutations of the same $r = 2^R$ numbers; that is, permutations on $\mathcal{P}$.

ii) at most $r = 2^R$ distinct permutations are present in all rows of all M_i.

Proof : By assumption, there are three 1:1, onto mappings

$\alpha : S \to \underline{S}$ (state assignement)

$\beta : X \to \underline{X}$ (input assignement)

$\gamma : Y \to \underline{Y}$ (output assignement)

from the state, input and output spaces of the S.S.M. onto those of the A.N.M., If the S.S.M. is an A.N.M., there also must be a 1:1 onto mapping, a "probability assignement".

$$\mu = P_\delta^{-1} : \mathcal{P} \to \underline{S} \ ;$$

i) hence, for all $\underline{s}_k \in \underline{S}$,

$$\text{Pr.}[\underline{s}_k/(\underline{s}_j,\underline{x}_i)] = \text{Pr.}[\delta_N(\underline{s}_j,\underline{x}_i) = \underline{s}_k] = \text{Pr.}[\delta(\underline{s}_j,\underline{x}_i) + n_\delta = \underline{s}_k] =$$

$$= \text{Pr.}[n_\delta = \underline{s}_k + \delta(\underline{s}_j,\underline{x}_i)] = P_\delta(\underline{s}_k + \delta(\underline{s}_j,\underline{x}_i)\ ).$$

For each $\underline{s}_k$, the difference $\underline{s}_k + \delta(\underline{s}_j,\underline{x}_i)^*$ is distinct and unique, and is one of the n_δ. Hence all of the $P_\delta(n_\delta)$ must appear in every row.

Q.E.D.

ii) First, it is proved that each distinct possible $\delta(\underline{s}_j,\underline{x}_i)$ must induce a distinct permutation of $\mathcal{P}$. Assume that $\delta_1(\underline{s}_j,\underline{x}_i)$ and $\delta_2(\underline{s}_j,\underline{x}_i)$ are distinct but produce the same permutation; this means that, for all $\underline{s}_k \in \underline{S}$,

$$P_\delta\ (\underline{s}_k + \delta_1(\underline{s}_j,\underline{x}_i)\ ) = P_\delta\ (\underline{s}_k + \delta_2(\underline{s}_j,\underline{x}_i)\ )\ ,$$

* For binary R-tuples, sum and difference coincide. If the field is not GF(2), a minus sign should be used.

or since all $P_\delta(n_\delta)$ are assumed distinct,

$$\delta_1(\underline{s}_j,\underline{x}_i) = \delta_2(\underline{s}_j,\underline{x}_i) ,$$

contradicting the hypothesis. Since there are at most r possible distinct next states for all pairs $(\underline{s}_j,\underline{x}_i)$, there must be at most r distinct permutations in the rows of all M_i^j. Q.E.D.

Let us now assume that the conditions of Theorem 3 are satisfied. Let the distinct permutations be arranged, in any row order, as a matrix M. The symbol $\hat{M}$ will denote the set of permutations of M, and this notation will be used throughout, for other matrices as well. The additive group table for $\underline{S}$, the rows of which result from adding each $\underline{s}_k \in \underline{S}$ to all vectors in $\underline{S}$, written in the standard order for R-tuples will be considered as an r x r matrix, or array, with entries in $\underline{S}$, and denoted by G_S.

Let us also assume, for the moment, that exactly r distinct permutations occur in the rows of the M_i. Then,

Theorem 4.

An S.S.M. is an A.N.M. (in its state transitions) iff there exist two orthogonal matrices C and D, the rows of which are of the form

$$[0,0,....,1,....] ,$$

with a single 1 per row and per column, such that, for some probability assignement μ

$$\mu\ (DM) = G_S\ C ,$$

where

$$[\mu(A)]_{i,j} = \mu(a_{ij}) ,$$

for any matrix A with entries a_{ij} in P.

Proof : If the S.S.M. is an A.N.M., then under a known probability assignment μ, and considering the proof of Theorem 3., it is seen that each row of M will go into a row of vectors in $\underline{S}$, obtained by adding a certain $\delta(\underline{s}_j\underline{x}_i)$ to each $s_k \in \underline{S}$. The order in which these s_k are found in the matrices M_i is not necessarily the standard one, as in G_S. Knowing the state assignement α, this non-standard order can be found, and C determined

Note that C changes only the positions of the columns of G_S. Also, since M is written in an arbitrary row order, the matrix D allows for their re-arrangement.

Conversely, if C,D and μ are given, the state assignement can be determined from C, and the probability distribution P_δ is simply μ^{-1}. The deterministic next states $\delta(\underline{s}_j,\underline{x}_i)$, for $\underline{s}_j \in \underline{S}$ and $\underline{x}_i \in \underline{X}$ are read from the M_i directly by finding the state $\underline{s}_k$ for which $P_\delta(\underline{s}_k \overline{+} \delta(\underline{s}_j,\underline{x}_i) = P_\delta(0)$, $0 \in \underline{S}$. Q.E.D.

Example of Application

Let us suppose that a binary input S.S.M. is given, and that from its matrix M the following M_i are obtained :

$$M_1 = \begin{vmatrix} c & h & f & d & a & b & g & e \\ a & e & g & b & c & d & f & h \\ a & e & g & b & c & d & f & h \\ b & g & e & a & d & c & h & f \\ c & h & f & d & a & b & g & e \\ a & e & g & b & c & d & f & h \\ e & a & b & g & h & f & d & c \\ c & h & f & d & a & b & g & e \end{vmatrix} \qquad M_2 = \begin{vmatrix} g & b & a & e & f & h & c & d \\ g & b & a & e & f & h & c & d \\ d & f & h & c & b & a & e & g \\ h & c & d & f & e & g & b & a \\ f & d & c & h & g & e & a & b \\ c & h & f & d & a & b & g & e \\ a & e & g & b & c & d & f & h \\ a & e & g & b & c & d & f & h \end{vmatrix}$$

where $P = \{a,b,\dots,h\}$ is any set of distinct numbers adding to 1.

From the M_i, M is written :

$$M = \begin{vmatrix} c & h & f & d & a & b & g & e \\ a & e & g & b & c & d & f & h \\ b & g & e & a & d & c & h & f \\ e & a & b & g & h & f & d & c \\ g & b & a & e & f & h & c & d \\ d & f & h & c & b & a & e & g \\ h & c & d & f & e & g & b & a \\ f & d & c & h & g & e & a & b \end{vmatrix}$$

and is seen to satisfy the conditions of Theorem 3. In this case also, exactly r = 8 distinct permutations occur. The following matrices C and D.

$$C = \begin{vmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{vmatrix} \qquad D = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{vmatrix}$$

and the probability assignement

$$\mu = \begin{pmatrix} a & b & c & d & e & f & g & h \\ 6 & 7 & 0 & 1 & 5 & 2 & 4 & 3 \end{pmatrix},$$

where the elements of $\underline{S}$ are denoted by their decimal equivalents (0=000, 1=001. 2=010, etc.) are such that

$$\mu(DM') = G_S C \quad ;$$

hence, by Theorem 4, the S.S.M. is an A.N.M.. The state assignement is read off the matrix C,

$$\alpha = \begin{pmatrix} t_1 & t_2 & t_3 & t_4 & t_5 & t_6 & t_7 & t_8 \\ 2 & 1 & 0 & 3 & 4 & 5 & 6 & 7 \end{pmatrix} \; ; \; t_j \in S \; ; \; 0,1,\dots \in \underline{S} \; ;$$

the deterministic next-state function is obtained from the M_i using $P_\delta(0) = C$,

$[\delta(s_j,x_i)]$	x_1	x_2
0	4	3
1	4	6
2	2	6
3	5	1
4	2	0
5	4	2
6	7	4
7	2	4

and, of course, P_8 is simply μ^{-1}. Note that the input assignement is arbitrary.

It is clear from the above developments that the row order of M, or that of G_S, are not important. They can be accounted for by the matrix D. Interpreting these matrices as sets of permutations on $\underline{S}$, Theorem 4 can be rewritten to include the cases in which less than r distinct permutations occur :

Theorem 5.

An S.S.M. is an A.N.M. (in its state transitions) iff there exists a probability assignement μ mapping the set $\hat{M}$ into a right coset of $\hat{G}_S$. That is, iff for some μ and a permutation c on $\underline{S}$,

$$(\mu\hat{M}) \subset \hat{G}_S \circ c$$

where $\circ$ indicates concatenation of permutations.

Proof : The expression of Theorem 4 may be rewritten

$$\mu M = D^{-1}\, G_S\, C.$$

when considered as sets of permutations on $\underline{S}$ (rows), the sets $(\widehat{\mu M})$ and $(\widehat{G_S C})$ must then be equal. Conversely, if $(\widehat{\mu M})$ and $(\widehat{G_S C})$ are equal, a matrix D can be found such that the above equality holds.

The postmultiplication of G_S by C changes its column order. This is the same as post-multiplying every permutation in G_S by a single permutation c. Since $\hat{G}_S$ is easily proved to be a subgroup of the group of all r ! permutations on $\underline{S}$, $G_S \circ c$ is one of its right cosets.

Even if less than r permutations occur in $\hat{M}$, the reasoning of theorems 3 and 4 holds, and the expression will read

$$(\widehat{\mu M}) \subset \hat{G}_S \circ c\ ,$$

which, of course, includes the case of equality. The appearance of less than r permutations in $\hat{M}$ implies only that the deterministic $\delta(\underline{s}_j,\underline{x}_i)$ is into $\underline{S}$ and not onto. Q.E.D.

The situation is actually somewhat simpler than Theorem 5 indicates. If μ is chosen arbitrarily so as to map one of the rows of M into the identity permutation on $\underline{S}$, say e, the expression above takes on a more familiar form :

Theorem 6.

Let μ^* be a probability assignement mapping any row of M into the identity permutation e on $\underline{S}$. Then the S.S.M. is an A.N.M. iff

$$(\mu^* M) = M^* = c^{-1} \circ \hat{G}_S \circ c$$

for some permutation c on $\underline{S}$. Inclusion must be used if less than r permutations occur in $\hat{M}$.

Proof : Assume that

$$(\widehat{\mu M}) = \hat{G}_S \circ c$$

But

$$(\widehat{\mu M}) = \mu\, \mu^{*-1}\, \mu^* \hat{M} = \tau^{-1} \circ M^*$$

since μ^* is invertible, and with $\tau^{-1} \triangleq \mu\, \mu^{*-1}$ being a permutation on $\underline{S}$.

Hence,

$$\mu \hat{M} = \hat{G}_S \circ c \Longleftrightarrow M^* = \tau \circ \hat{G}_S \circ c$$

for some τ. M^*, however, contains $e \in \hat{G}_S$; hence, for some $g \in \hat{G}_S$,

$$e = \tau \circ g \circ c \text{ or } \tau = c^{-1} \circ g^{-1}$$

and finally,

$$\tau \circ \hat{G}_S \circ c = c^{-1} \circ g^{-1} \circ \hat{G}_S \circ c = c^{-1} \circ \hat{G}_S \circ c$$

Conversely, if $\mu^* \hat{M} = c^{-1} \circ \hat{G}_S \circ c$

for some c, then, $c \circ \mu^* \hat{M} = \hat{G}_S \circ c$

and $\mu \hat{M} = \hat{G}_S \circ c$ with $\mu = c \circ \mu^* : P \to \underline{S}$. All the reasoning above applies to the case in which less than r permutations occur in M. Q.E.D.

4. CONCLUSIONS

The expression $M^* = c^{-1} \circ \hat{G}_S \circ c$ indicates that M^* must be a group, conjugate and isomorphic to G_S [11]. The problem of finding c, if it exists, is reduced to the problem of finding a possible isomorphic between the two groups. The authors are developing an algorithm for this purpose. It also seems that all the above results are extendable to include the outputs.

Based on all the results presented in this paper, plus additional results of the authors on the different notions of equivalence between non-deterministic machines, the statement can be made that the S.S.M. is, by far, the most general and the more promising of all models studied so far. It includes all others as particular cases, and may function, as the authors will show elsewhere, as a sequence recognizer with perhaps still some advantage over minimal M.P.A.'s. We venture that a fully unified theory of non-deterministic finite-state systems can be built, using only the S.S.M. as a general, "abstract", model and assigned S.S.M.'s (such as the A.N.M.) as more physical, sequential-circuit oriented models.

5. REFERENCES

[1] Rabin, M.O., "Probabilistic Automata" on Sequential Machines : Selected Papers, E.F. Moore ed., Addison-Wesley Publ. Co., Reading, Mass., 1964. pp. 98-114.

[2] Paz, A., "Some Aspects of Probabilistic Automata", Information and Control, Vol. 9, No. 1, February 1966, pp. 26-60.

[3] Salomaa, A., "On m-Adic Probabilistic Automata", Information and Control, Vol. 10, 1967, pp. 215-219.

[4] Carlyle, J.W., "Equivalent Stochastic Sequential Machines", Tech. Report Series 60, Issue 415, Univ. of California at Berkeley. Also as "Reduced Forms for Stochastic Sequential Machines", J. Math. Anal. & Applic. Vol. 7, 1963, pp. 167-175.

[5] Bacon, G.C. "Minimal-state Stochastic Finite-State Systems", I.E.E.E. Trans. on Circuit Theory, (Correspondence), Vol. CT-11, June, 1964.

[6] Ott, G.H., "Reconsider the State-Minimization Problem for Stochastic Finite-State Systems", I.E.E.E. Conf. Record, 7th Ann. Symp. on Switching and Automata Thy., Berkeley, California, October 1966.

[7] Eaton, J.H., and Zadeh, L.A., "Optimal Pursuit Strategies in Discrete-State Probabilistic Systems", Journal of Basic Engineering, Trans. of the A.S.M.E., Series D, Vol. 84, 1962, pp. 23-29.

[8] Dorato, P. Drenick R.F., and Shaw, L., Optimal Stochastic Control Theory : a Short Course, Dept. of Electrical Engineering, Pol. Inst. of Brooklyn, New York, 1964, Chapter 2.

[9] Kashyap, R.L., "Optimization of Stochastic Finite-State Systems", I.E.E.E. Trans. on Auto Control, Vol. AC-11, No. 4, October, 1966, pp. 685-692.

[10] Ash, R. Information Theory, Interscience Publishers, John Wiley & Sons, New York, 1965.

[11] Carmichael, R.D., Introduction to the Theory of Groups of Finite Order, Dover Publications Inc., New York, 1956.

ON THE PERIOD AND CHARACTERISTIC EQUATION OF AN AUTONOMOUS F-CHANNEL PARALLEL LINEAR FEEDBACK SHIFT REGISTER

M. Y. HSIAO
Department of Electrical Engineering
University of Florida
Gainesville, Florida

ABSTRACT

This paper discusses the periods of a f-channel parallel linear feedback shift register or the f-channel analogy circuit, can be easily converted from its original serial linear feedback shift register. Three converting theorems are given. Next, three methods of obtaining the characteristic equation of a parallel circuit are shown. Finally, a comparison between various approaches of finding the cycle set of a f-channel analogy circuit is given.

I. Introduction

A f-channel parallel linear feedback shift register (LFSR),[1] or a f-channel analogy circuit,[2] is a LFSR with f-input and f-output characterized by a polynomial g(x) of degree r such that it performs the same operation as its original serial LFSR specified by g(x) but in a speed f-time faster.

Periods of an autonomous serial LFSR are solved and may be referred to papers in a book edited by Kautz.[3] Since all cycles in a LFSR are closed and branchless, then the following three obvious converting theorems can be applied to any cycle as well.

These three theorems are stated in Section II. Section III discusses three methods of finding the characteristic equation of an autonomous f-channel parallel LFSR. Section IV summarizes the results and points out the periods of a f-channel parallel LFSR are not always uniquely specified by its characteristic equation which is the same as that of the serial LFSR.

II. The Three Converting Theorems

The periods of a f-channel parallel LFSR, denoted by P_p, can be easily obtained from the periods of the original serial LFSR, denoted by P_s, using a set of converting rules. Since any serial LFSR has more than one isolated state transition path, it has at least two cycles which forms a cycle set. The number of states in each closed circular path is defined as the period of this cycle.

Definition 1: The set all possible cycles is called the cycle set of the LFSR or of the companion matrix T.

Here we let

$$g(x) = g_o + g_1x + g_2 x^2 + \dots + g_r x^r \qquad (1)$$

then

$$T = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ & & \vdots & & & \\ 0 & 0 & & & \dots & 01 \\ g_o & g_1 & & \cdot & \cdot\ \cdot & g_{r-1} \end{bmatrix} \qquad (2)$$

Notation: Cycle set = $\{\mu_1(k_1), \dots, \mu_n(k_n)\}$ where $\mu_i(k_i)$ implies μ_i cycles of length k_i.

Definition 2: The period P_s of a serial LFSR is defined as the smallest integer for which $T^{P_s} = I$, where I is the identity matrix. Therefore we shall assume that periods of a given serial LFSR are known[3][5] and we try to establish the link between each P_s and its corresponding P_p. The result deals with one cycle, but because all cycles are isolated to each other, the result can certainly be applied to other cycles as well.

Theorem 1: If $P_s = mf$, then $P_p = m$ where m is a positive integer.

Proof: By definition 2,

$$T^{P_s} = I$$

but $P_s = mf = T^{mf} = I = [T^f]^m = I$

again by the definition 2,

$$[T^f]^{P_p} = I$$

$$\therefore m = P_p$$

Theorem 2: If f and P_s are relative prime, then $P_p = P_s$.

Proof: $$T^{P_s} = I \qquad (3)$$

and $$[T^f]^{P_p} = I \qquad (4)$$

Comparing equations (3) and (4), with the fact that $I^k = I$ where k is a positive integer, we have

$$f P_p = kP_s$$

or $$P_p = k \frac{P_s}{f} \qquad (5)$$

Since P_s and f are relative prime, then the smallest k that satisfies the equation (5) is k equal to f for which P_p is an integer. Therefore, we have

$$P_p = P_s$$

Theorem 3: If f and P_s have a common factor $c > 1$, i.e., $f = ca$ and $P_s = cb$ where a and b are positive integers and are relative prime, then $P_p = \frac{P_s}{c}$

Proof: From the proof of the theorem 2, we have

$$P_p = k \frac{P_s}{f} = k \frac{cb}{ca} \qquad (6)$$

The smallest integer k to make P_p on integer in the equation (6) is k equal to a. Therefore, we have

$$P_p = \frac{P_s}{c}$$

The above three theorems have established the link between one period of a given serial LFSR to the corresponding period of its f-channel analogy parallel LFSR. Therefore all periods of a parallel LFSR can be found based on known periods of its original serial LFSR.

III. Methods of finding the characteristic equation of an autonomous f-channel parallel LFSR

The cycle set structure of an autonomous serial LFSR is completely specified by its characteristic polynomial.

Definition 3: The characteristic equation $\Phi_f(\lambda) = 0$ of a rxr matrix T^f is the determinant $|T^f - \lambda I| = 0$, where the determinant $|T^f - \lambda I| = \Phi_f(\lambda)$ is the characteristic polynomial.

Definition 4: The minimal polynomial T is the monic polynomial $m(\lambda)$ (coefficient of the highest power term equals to 1) of lowest degree such that $m(T) = 0$.

Therefore, $m_f(\lambda) \,|\, \Phi_f(\lambda)$ and in general, $m(\lambda) \neq \Phi(\lambda)$

Theorem 4: In the serial case, $f = 1$, the characteristic and minimum polynomial of the companion matrix of $g(x)$ are both equal to $g(x)$, i.e.,

$$g(\lambda) = \Phi_1(\lambda) = m_1(\lambda)$$

Proof: See p.148 of Perlis.[6]

The general procedure of finding $\Phi_f(\lambda)$ of an autonomous f-channel parallel LFSR is diagrammed as follows:

$$g(x) \rightarrow T \rightarrow T^f \rightarrow \Phi_f(\lambda)$$

Various methods of finding $\Phi_f(\lambda)$ are discussed below.

(i) Direct method: By the definition 3.

$$\Phi_f(\lambda) = |T^f - \lambda I| \tag{7}$$

We shall first find T^f and then evaluate the determinant $|T^f - \lambda I|$ directly.

Theorem 5: [6] (Cayley-Hamilton) Every square matrix satisfies its own characteristic polynomial, i.e.,

$$\Phi_f(T^f) = 0$$

Example 1: Given $g(x) = 1 + x + x^4$, find $\Phi_3(\lambda)$.

Solution:

$$g(x) = T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \Rightarrow T^3 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\Rightarrow \Phi_3(\lambda) = |T^3 - \lambda I| = \begin{vmatrix} \lambda & 0 & 0 & 1 \\ 1 & 1+\lambda & 0 & 0 \\ 0 & 1 & 1+\lambda & 0 \\ 0 & 0 & 1 & 1+\lambda \end{vmatrix} = \lambda^4 + \lambda^3 + \lambda^2 + \lambda + 1$$

The Cayley-Hamilton theorem can be easily verified as $\Phi_3(T^3) = 0$

The direct method of finding a characteristic equation becomes difficult if the dimension of T^f is large.

Several alternative methods are given below, mostly based on the work of Albert.[4]

(i) Polynomial root method: This method provides an easy way of finding T^j based on the table of all powers of a root α^{+} of $g(x)$. Note that this method is different from the method of polynomial root approach given previously by Hsiao and Sih.[1]

Since all elements in $GF(2^r)$ forms a r-dimensional vector space with a set of linear independent basis as $1, \alpha, \alpha^2, \ldots, \alpha^{r-1}$ where α^j $(i = 0,1,2,\ldots,r-1)$ is a binary r-tuple defined as follows.

$$\begin{aligned} \alpha^0 &= \overbrace{1\ 0\ 0\ \ldots\ 0\ 0}^{r} \\ \alpha^1 &= 0\ 1\ 0\ \ldots\ 0\ 0 \\ &\vdots \\ \alpha^{r-1} &= 0\ 0\ 0\ \ldots\ 0\ 1 \end{aligned} \tag{9}$$

Then any element $\alpha^j \in GF(2^r)$ can be expressed as

$$\alpha^j = \xi_{11} + \xi_{12}\,\alpha + \xi_{13}\,\alpha^2 + \ldots + \xi_{1r}\,\alpha^{r-1}$$

with $\xi_{ij} \in GF(2)$.

Next, compute the product $\alpha^{j+i-1} = \xi_{i1} + \xi_{i2}\,\alpha + \ldots + \xi_{ir}\,\alpha^{r-1}$ (10)

for $i = 1, 2, 3, \ldots, (r-1)$ and then

$$T^j = [\xi_{ij}]_{rxr} = \begin{bmatrix} \alpha^j \\ \alpha^{j+1} \\ \vdots \\ \alpha^{j+r-1} \end{bmatrix}_{rxr} \tag{11}$$

Equation 11 can be verified through induction.

+ If $g(x)$ has the period P_s, then $g(x) \mid x^{P_s}-1$. The root α of $g(x)$ is then a P_s-th root of unity.

$$k = 0,\ T^k = T^o = I_{rxr} = \begin{bmatrix} \alpha^o \\ \alpha^1 \\ \vdots \\ \alpha^{r-1} \end{bmatrix}$$

$$k = 1,\ T^k = T^1 = T = \begin{bmatrix} \alpha^1 \\ \alpha^2 \\ \vdots \\ \alpha^{r-1} \\ \alpha^r \end{bmatrix} \tag{12}$$

Since α is also the residue class of x as defined by Peterson[7] such that

$$\alpha \equiv \{x\}$$

Therefore, with $g_r = 1$,

$$\alpha^r \equiv g_o\ \alpha^o + g_1\ \alpha + \ldots + g_{r-1}\ \alpha^{r-1}$$

i.e.,

$$\alpha^r = (g_o, g_1, \ldots, g_{r-1}) \tag{13}$$

Substituting the Equation (13) into the Equation (12), we have exactly the companion matrix T. Next, assume k = j is true, i.e.,

$$T^j = \begin{bmatrix} \alpha^j \\ \alpha^{j+1} \\ \vdots \\ \alpha^{j+r-1} \end{bmatrix}$$

we shall show that k = j+1 is also true.

$$T^{j+1} = T \cdot T^j = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & 0 \ldots & 0 \\ & & \vdots & & \\ 0 & 0 & \vdots & \ldots\ldots & 1 \\ g_o & g_1 & & \ldots\ldots & g_{r-1} \end{bmatrix} \begin{bmatrix} \alpha^j \\ \alpha^{j+1} \\ \vdots \\ \alpha^{j+r-1} \end{bmatrix}$$

$$= \begin{bmatrix} \alpha^{j+1} \\ \alpha^{j+2} \\ \vdots \\ \alpha^{j+r-1} \\ g_o\ \alpha^j + g_1\ \alpha^{j+1} + \ldots + g_{r-1}\ \alpha^{j+r-1} \end{bmatrix}$$

but
$$\begin{aligned} & g_o\ \alpha^j + g_1\ \alpha^{j+1} + \ldots + g_{r-1}\ \alpha^{j+r-1} \\ &= \alpha^j\ (g_o + g_1\ \alpha + \ldots + g_{r-1}\ \alpha^{r-1}) \\ &= \alpha^j\ \ \alpha^r = \alpha^{r+j} \end{aligned}$$

Therefore, the last row of the matrix T^{j+1} is equal to α^{r+j} and the proof is thus completed.

Example 2: Given $g(x) = 1 + x + x^4$ and let α be a root of $g(x)$, then any power of T can be found from the following Table 1 which lists all power of α.

Table 1 All Powers of α

α^o	1 0 0 0
α^1	0 1 0 0
α^2	0 0 1 0
α^3	0 0 0 1
α^4	1 1 0 0
α^5	0 1 1 0
α^6	0 0 1 1
α^7	1 1 0 1
α^8	1 0 1 0
α^9	0 1 0 1
α^{10}	1 1 1 0
α^{11}	0 1 1 1
α^{12}	1 1 1 1
α^{13}	1 0 1 1
α^{14}	1 0 0 1

Then
$$T^j = \begin{bmatrix} \alpha^j \\ \alpha^{j+1} \\ \alpha^{j+2} \\ \alpha^{j+3} \end{bmatrix}$$

Suppose $j = 8$, then
$$T^8 = \begin{bmatrix} \alpha^8 \\ \alpha^9 \\ \alpha^{10} \\ \alpha^{11} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

The above method can be used to find T^{-j} for all j. Let the period of cycle which contains α be P_1, then $\alpha^{P_1} = 1$. Therefore,

$$T^{P_1} \Rightarrow I = T^j \, T^{P_1 - j} = I \Rightarrow T^{-j} = T^{P_1 - j}$$

The third method to be described below utilizes the tables in the Appendix C of Peterson and thus restricted itself to a certain class of polynomials.

(iii) Minimum polynomial method: Here the definition of minimum polynomial is given again based on the polynomial root rather than the T matrix.

Definition 5: Let β be any element of $GF(2^r)$, the monic polynomial $m(x)$ of smallest degree over $GF_2[x]$* such that $m(\beta) = 0$ is called the minimum polynomial.

If $g(x)$ is a primitive polynomial and α be its primitive root, then

$$\Phi_f(\lambda) = [m_f(\lambda)]^C$$

where $m_f(\lambda)$ is the minimum polynomial of α^f with degree m and r=mc.

Proof: The proof is almost the same as that of Theorem 4.8 of Albert (P100).[4]

Example 3: Let α be a primitive root of $g(x) = 1 + x + x^4$, then from p.254 of Peterson, we find

$$m_1(x) = 1 + x + x^4$$

$$m_3(x) = 1 + x + x^2 + x^3 + x^4$$

$$m_5(x) = 1 + x + x^2$$

Therefore,

$$\Phi_1(\lambda) = [m_1(\lambda)]^1 = 1 + \lambda + \lambda^4$$

$$\Phi_3(\lambda) = [m_3(\lambda)]^1 = 1 + \lambda + \lambda^2 + \lambda^3 + \lambda^4$$

$$\Phi_5(\lambda) = [m_5(\lambda)]^2 = (1 + \lambda + \lambda^2)^2 = 1 + \lambda^2 + \lambda^4$$

The above result can be easily verified through other methods. In GF(2), as derived from the Theorem 6.26 of Peterson, the following is true.

$$m_i(x) = m_{2i}(x) = m_{2^2 i}(x) = \dots = m_{2^j i \bmod (2^r-1)}(x) = \dots$$

$$= m_{2^{m-1} i \bmod (2^r-1)}(x)$$

---

*$GF_2[x]$ is the polynomial domain containing all polynomials in x with coefficients in CF(2).

Therefore in Example 4, we have

$$\Phi_1(\lambda) = \Phi_2(\lambda) = \Phi_4(\lambda) = \Phi_8(\lambda)$$

$$\Phi_3(\lambda) = \Phi_6(\lambda) = \Phi_{12}(\lambda) = \Phi_9(\lambda)$$

$$\Phi_5(\lambda) = \Phi_{10}(\lambda)$$

The last result can be stated in another way, i.e.,
If T is associated with g(x) and $\Phi(T) = 0$, then $\Phi(T^2) = \Phi(T^{2^2}) = \ldots = \Phi(T^{2^{m-1}}) = 0$.

DISCUSSION

It is natural to ask whether the decomposition of the characteristic equqtion of an autonomous f-channel LFSR obtained by methods of Section III will give the same cycle set as the result of using those three theorems of Section II. From the author's test on various examples, only one example does not work exactly as it is expected. Table 2 summarizes all test results of examples in this paper. It indicates there exists at least one disagreed example. The reason is that the polynomial $\Phi_5(x)$ contains repeated factors, a case pointed out by Elspas[3] that the circuit behavior is not uniquely determined by the characteristic polynomial. In conclusion, the cycle set of an autonomous parallel LFSR is best evaluated from the result of Section II.

ACKNOWLEDGEMENT

The author wishes to thank Dr. W. H. Chen and Dr. T. S. George for their encouragement.

REFERENCES

1. M. Y. Hsiao and K.Y. Sih, "Serial-to-parallel transformation of linear feedback shift register circuits" IEEE Trans. on Electronic Computers, Dec. 1964.
2. Gill, A., "On the serial-to-parallel transformation of linear sequential circuits" IEEE Trans. on Electronic Computers, Feb. 1966.
3. Kautz, W.H., editor "Linear sequential switching circuits" Holden Day Inc. 1965, a paper by Elspas.
4. Albert, A.A., "Fundamental concepts of higher algebra" University of Chicago Press, 1956.
5. Fitzpatrick, G.B., "Synthesis of binary ring counters of given period" J. ACM, July 1960.
6. Perlis, S., "Theory of matrices" Addison-Wesley Co., 1958.
7. Peterson, W.W., "Error correcting codes" MIT Press 1962.

TABLE 2

$\Phi_1(x)$	Cycle set (c.s.) of $\Phi_1(x)$	Results of Section II Cycle set of $\Phi_1(x)$ Transformed to f-channel	Result of Section III $\Phi_f(x) = \lvert T^f - xI \rvert$	Does the results of §II & §III agree by way of Ref. 3 & 5?
$1+x+x^4$	$\{1(1),\ 1(15)\}$	f=3 $\{1(1),\ 3(5)\}$	$\Phi_3(x) = 1+x+x^2+x^3+x^4$ c.s.= $\{1(1),\ 3(5)\}$	Yes
$1+x+x^4$	$\{1(1),\ 1(15)\}$	f=5 $\{1(1),\ 5(3)\}$	$\Phi_5(x) = 1+x^2+x^4$ c.s.= $\{1(1),\ 1(3),\ 2(6)\}$	No *
$1+x^2+x^5$	$\{1(1),\ 1(31)\}$	f=2,4 $\{1(1),\ 1(31)\}$	$\Phi_2(x) = \Phi_4(x) = 1+x^2+x^5$ c.s.= $\{1(1),\ 1(31)\}$	Yes
$1+x^2+x^5$	$\{1(1),\ 1(31)\}$	f=3 $\{1(1),\ 1(31)\}$	$\Phi_3(x) = 1+x^2+x^3+x^4+x^5$ c.s.= $\{1(1),\ 1(31)\}$	Yes
$1+x^2+x^5$	$\{1(1),\ 1(31)\}$	f=3 $\{1(1),\ 1(31)\}$	$\Phi_5(x) = 1+x^2+x^3+x^5$ c.s.= $\{1(1),\ 1(31)\}$	Yes

* See ref. 3

STARTING STATE GENERATING FUNCTION FOR AUTONOMOUS P-ARY FEEDBACK SHIFT REGISTER

FREDERIC J. MOWLE
School of Electrical Engineering
Purdue University
Lafayette, Indiana

ABSTRACT

Several analytical techniques are given for analyzing p-ary nonlinear feedback shift registers. In particular, the generating function for the set of all starting states on a nonlinear p-ary FSR is obtained directly in terms of its feedback function.

INTRODUCTION

At past Allerton Conferences, papers have been presented which discussed analytical and graphical techniques for determining the stability of non-linear feedback shift registers under autonomous operation. [1,2,3] In this paper analytical techniques will be given for analyzing a nonlinear p-ary feedback shift register in terms of its feedback function. In particular a simple method for determining the set of all states which have no predecessors will be described. The method used will lead to a generating function for the set of all starting states in terms of the feedback function of the shift register. Knowing the complete set of starting states allows one to systematically determine the set of all branch states and the set of all closed cycles.

Let $F(\underline{s})$ be the feedback function of a p-ary autonomous feedback shift register of the type shown in Figure 1.

This work was supported by the National Science Foundation under Grant GK-1542.

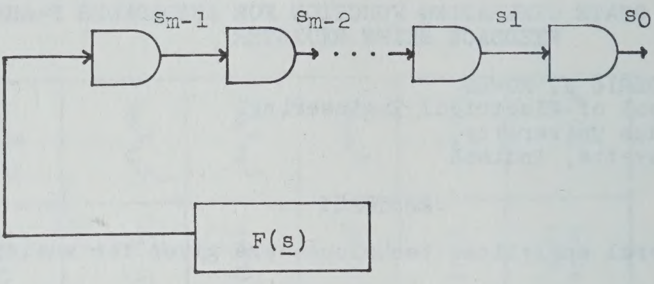


FIGURE 1. Autonomous p-ary Feedback Shift Register

The state of the machine will be denoted by the vector $\underline{s} = (s_{n-1}, \ldots, s_1, s_0)$ where each s_j is the output of one of the m delay stages. If $\underline{s}$ denoted the present state of the machine, then $\theta\underline{s}$ will denote the next state of the machine and $\theta^i\underline{s}$ will denote the state of the machine after i shifts. "θ" will be called the next state operator. Thus if $\underline{s} = (s_{m-1}, s_{m-2}, \ldots, s_1, s_0)$ $\theta\underline{s} = (F(\underline{s}), s_{m-1}, s_{m-2}, \ldots, s_2, s_1)$ where each s_j and the feedback function $F(\underline{s})$ can assume any value in the Galois Field p, GF(p) . The feedback function $F(\underline{s})$ of an m stage machine can assume any of the possible p^{p^m} distinct functions of the m variables.

The function $G_A(\underline{s})$ is said to generate the subset of states A on a machine M if

$$G_A(\underline{s}) = 1 \qquad \underline{s} \in A$$
$$G_A(\underline{s}) = 0 \qquad \underline{s} \notin A$$

We will use $G_A(\underline{s})$ to denote the generating function of set A.

Example 1. Consider the two stage ternary machine shown in Figure 2. The generating function for the set of states $A = \{11,20,21\}$ is given by

$$G_A(s_1,s_0) = (s_1^2+s_1)(s_0^2+s_0) + (2s_1^2+s_1)(s_0^2+2s_0+1)$$

(+ denotes addition module p)

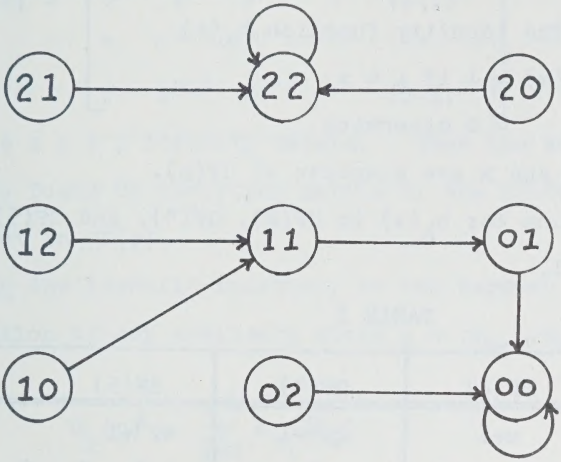


FIGURE 2. State Diagram of a Ternary FSR

Definition 1. The Predecessor Generating Function. If $G_A(\underline{s})$ generates the set A on machine M, then $G_A(\theta\underline{s})$ will generate the set of all states B such that $\theta B \subseteq A$. $G_A(\theta\underline{s})$ will be called the predecessor generating function.

Using our previous example, the set B of all predecessor states of the set A = {11,20,21} is generated by

$$G_B(\underline{s}) = G_A(\theta\underline{s}) = (F(\underline{s})^2 + F(\underline{s}))\ (s_1^2 + s_1) +$$

$$(2F(\underline{s})^2 + F(\underline{s}))\ (s_2^1 + 2s_1 + 1)$$

$$= 2(s_1^2 + s_1)\ (s_0^2 + s_0 + 1)$$

B = {10,12}

where the feedback function for the machine in Figure 2 is given by $F(\underline{s}) = 2s_1^2s_0^2 + s_1^2s_0 + 2s_1s_0^2 + s_1s_0 + 2s_1$

Before we can conveniently define a successor generating function, a few additional analytical tools will prove helpful.

Definition 2. The identity function $h_\lambda(x)$

$$h_\lambda(x) = 1 \text{ if } \lambda = x$$
$$= 0 \text{ otherwise}$$

where λ and x are elements of $GF(p)$.

The equations for $h_\lambda(x)$ in $GF(2)$, $GF(3)$, and $GF(5)$ are given in Table 1.

TABLE I

	GF(2)	GF(3)	GF(5)
$h_0(x)$	$x+1$	$2x^2+1$	$4x^4+1$
$h_1(x)$	x	$2x^2+2x$	$4x^4+4x^3+4x^2+4x$
$h_2(x)$		$2x^2+x$	$4x^4+3x^3+x^2+2x$
$h_3(x)$			$4x^4+2x^3+x^2+3x$
$h_4(x)$			$4x^4+x^3+4x^2+x$

In $GF(p)$ we can write a set of p simultaneous algebraic equations for $h_\lambda(x)$

$$h_\lambda(x) = \sum_{i=0}^{p-1} a_{\lambda i} x^i = \delta_{\lambda x}, \quad 0 \leq \lambda \leq p-1$$

We can also express the complete set of equations for all λ's by the following matrix equation

$$[A] \cdot [B] = [I]$$

where

$$[A] = \begin{bmatrix} a_{0\,0} & - & - & - & a_{0\,p-1} \\ - & & & & - \\ - & & & & - \\ a_{p-1\,0} & - & - & - & a_{p-1\,p-1} \end{bmatrix}$$

$$[B] = \begin{bmatrix} 1 & 1 & 1 & \cdot & 1 \\ 0 & 1 & 2 & \cdot & p-1 \\ 0 & 1^2 & 2^2 & \cdot & (p-1)^2 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 1^{p-1} & 2^{p-1} & \cdot & (p-1)^{p-1} \end{bmatrix}$$

and $[I]$ is a p x p Identity Matrix. Thus the coefficients $a_{\lambda j}$ can be found by inverting matrix B, the arithmetic being carried out in GF(p).

Using the identity function, we can express the generating function of any arbitrary state $\underline{c} = (c_{m-1}, c_{m-2}, \ldots, c_1, c_0)$ as

$$G_{\underline{c}}(\underline{s}) = \prod_{i=0}^{m-1} h_{c_i}(s_i) \qquad (1)$$

Since each state on the feedback shift register can be uniquely expressed in this manner, we can write the generating function for an arbitrary set of states on M as a finite sum of terms of the form given in (1). Also, since each state feeds back a fixed value, we can express the feedback function of the machine in a sum of product canonic form as

$$F(\underline{s}) = \sum_{j=0}^{p^m-1} K_{\underline{c}_j} \prod_{i=0}^{m-1} h_{c_{ji}}(s_i)$$

where we have numbered each state $\underline{c}_i$ as follows

$\underline{c}_0 = (0,0,0,\ldots, 0)$, $\underline{c}_1$ $(0,0,0,\ldots, 0\ 1)$

$\underline{c}_p = (0,0,0,\ldots, 1\ 0) \cdots \underline{c}_{p^m-1} = (p-1,p-1,\ldots, p-1,p-1)$

and $K_{\underline{c}_j}$ is the feedback coefficient for state $\underline{c}_j$ and c_{ji} is the ith's component of state $\underline{c}_j$.

Definition 3. If $\underline{s} = (s_{m-1}, s_{m-2}, \ldots, s_1, s_0)$ is the present state of a machine M, then $L_a\underline{s}$ will denote the state $(s_{m-2}, \ldots, s_1, s_0, a)$. The state obtained by left shifting all the components one position and inserting "a" in the right most position, the s_{m-1} component being lost in the process, "L_a" will be called the left shifting operator.

Let $\underline{c} = (c_{m-1}, c_{m-2}, \ldots, c_1, c_0)$ be an arbitrary state on M and let $F(\underline{c}) = \lambda$, i.e. $\underline{c}$ is a λ feedback state. Then $\theta\underline{c} = (\lambda, c_{m-1}, \ldots, c_2, c_1)$ is the next state on the machine after $\underline{c}$. Analytically we can express this using the generating function as

$$G_{\underline{c}}\ (\underline{s}) = \prod_{i=0}^{m-1} h_{c_i}(s_i)$$

c_j being the j-th component of $\underline{s}$ and

$$\theta G_{\underline{c}}\ (\underline{s}) = h_\lambda(s_{m-1}) \prod_{i=0}^{m-2} h_{c_{i+1}}(s_i) \tag{2}$$

Since in general an arbitrary set of states on M will not have a generating function expressed in such a convenient sum of product form, we would like to find an alternate form for expressing $\theta G_{\underline{c}}\ (\underline{s})$.

One such form is the following

$$\begin{aligned}\theta G_{\underline{c}}\ (\underline{s}) &= h_\lambda(s_{m-1}) \left[1 + \prod_{k=0}^{p-1} \left[G_{\underline{c}}(L_k\underline{s}) + p-1 \right] \right] \\ &= h_\lambda(s_{m-1}) \left[1 + \prod_{k=0}^{p-1} \left[h_{c_0}(k) \prod_{i=0}^{m-2} \left[h_{c_{i+1}}(s_i) + p-1 \right] \right] \right]\end{aligned} \tag{3}$$

With $\underline{c} = (c_{m-1}, c_{m-2}, \ldots, c_1, c_0)$, $G_{\underline{c}}(L_k\underline{s})$ will be nonzero only for $k = c_0$, since $h_{c_0}(k) = 1$ iff $c_0 = k$.

Thus we can rewrite (3) as

$$\theta G_{\underline{c}}(\underline{s}) = h_\lambda(s_{m-1})\left[1 + (p-1)^{p-1}\prod_{i=0}^{m-2}\left[h_{c_{i+1}}(s_i) + (p-1)^{p-1}\right]\right]$$

but $(p-1)^{p-1} = 1$ and $(p-1)^p = p-1$

$$\therefore \quad \theta G_{\underline{c}}(\underline{s}) = h_\lambda(s_{m-1})\prod_{i=0}^{m-2} h_{c_{i+1}}(s_i)$$

Thus the two descriptions of $\theta G_{\underline{c}}(\underline{s})$ are the same.

Definition 4. The Successor Function.

If $G_A(\underline{s})$ generates a set A then $\theta G_A(\underline{s})$ generates the set $\mathbf{C}$ such that $\theta A = \mathbf{C}$, i.e. all states in A have successors in $\mathbf{C}$ and $\theta G_A(\underline{s})$ will be called the successor generating function.

$$\theta G_A(\underline{s}) = \sum_{i=0}^{p-1}\left[G_{A_i}(\underline{s})\right]\Big|_i$$

where $G_{A_i}(\underline{s}) = G_A(\underline{s})\, h_i(F(\underline{s}))$ is the generating function for all states in A which have the feedback function $F(\underline{s}) = i$. The operator $[\ ]|_i$ effectively shifts the components of each state right one position and inserts the feedback constant i in the left most position. Thus we have

$$\theta G_A(\underline{s}) = \sum_{i=0}^{p-1} h_i(s_{m-1})\left[1 + \prod_{k=0}^{p-1}\left[G_{A_i}(L_k\underline{s}) + p-1\right]\right]$$

with $G_{A_i}(\underline{s}) = G_A(\underline{s})\, h_i(F(\underline{s}))$.

Although generally the equation is quite cumbersome to handle; in the Binary case, we can express $\theta G_A(\underline{s})$ as

$$\theta G_{\underline{c}}(\underline{s}) = (s_{m-1})\left[G_{A_0}(L_0\underline{s}) \cdot G_{A_0}(L_1\underline{s}) + G_{A_0}(L_0\underline{s}) + G_{A_0}(L_1\underline{s})\right]$$
$$+ (s_{m-1}+1)\left[G_{A_1}(L_1\underline{s}) \cdot G_{A_1}(L_1\underline{s}) + G_{A_1}(L_0\underline{s}) + G_{A_1}(L_1\underline{s})\right]$$

Note that $h_0(x) = x+1$ and $h_1(x) = 1$ in the binary case.

Let us now consider the problem of finding which states in set A on a machine M have no predecessors (the set of starting states). One method that can be used is the following. Start with the generating function of Set A. Calculate $G_A(\theta\underline{s})$, the set of states which are all the predecessors of the states in A. Next, calculate $\theta G_A(\theta\underline{s})$ and subtract this from $G_A(\underline{s})$, leaving the desired result. Thus if we denote the set of starting states by T_A we have that

$$\begin{aligned} G_{T_A}(\underline{s}) &= G_A(\underline{s}) - \theta G_A(\theta\underline{s}) \\ &= G_A(\underline{s}) + (p-1)\,\theta G_A(\theta\underline{s}) \end{aligned}$$

If we are interested in the complete set of starting states, then we will choose the set A to be the set M, the set of all states on the machine. Thus, we have $G_M(\underline{s}) = 1$ and $G_M(\theta\underline{s}) = 1$ and $G_T(\underline{s}) = 1 + (p-1)\theta(1)$.

$$G_T(\underline{s}) = 1 + (p-1) \sum_{i=0}^{p-1} h_i(s_{m-1}) \left[1 + \prod_{k=0}^{p-1} \left[G_{M_i}(L_k\underline{s}) + p-1 \right] \right].$$

$$\text{where } G_{M_i}(\underline{s}) = h_i\left[F(\underline{s})\right]$$

For the binary case (p=2) this equation can be reduced to

$$\begin{aligned} G_T(\underline{s}) &= 1 + (s_{m-1} + 1)\left[1 + F(L_0\underline{s})\cdot F(L_1\underline{s})\right] + \\ &\qquad (s_{m-1})\left[1 + (F(L_0\underline{s}) + 1)\cdot(F(L_1\underline{s}) + 1)\right] \\ &= s_{m-1}\left[F(L_0\underline{s}) + F(L_1\underline{s}) + 1\right] + F(L_0\underline{s})\,F(L_1\underline{s}). \end{aligned}$$

For example, if we let $F(\underline{s})$ be a binary monotone FSR, then

$F(\underline{s}) = s_0 H(s_{m-1}, \ldots, s_1)$ and $F(L_0\underline{s}) = 0$.

$F(L_1\underline{s}) = H(s_{m-2}, \ldots, s_0)$ and $G_T(\underline{s}) = (s_{m-1})\left[H(s_{m-2}, \ldots, s_0) + 1\right]$

Thus on the FSR with $F(\underline{s}) = s_0(s_1 + s_2)$ we have that

$G_T(\underline{s}) = s_2\,(s_1 + s_0 + 1) \quad T = \{100, 111\}$

The state diagram for this machine is shown in Figure 3.

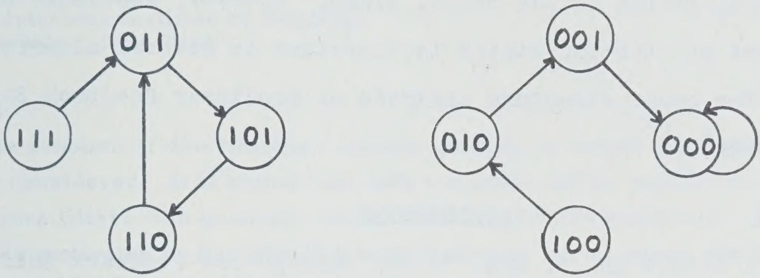


FIGURE 3. Binary FSR with $F(\underline{s}) = s_0(s_1 + s_2)$

Next let us consider linear binary FSR's of the form

$$F(\underline{s}) = \sum_{i=0}^{m-1} a_i s_i$$

$$F(L_0\underline{s}) = \sum_{i=0}^{m-2} a_{i+1} s_i$$

$$F(L_1\underline{s}) = a_0 + \sum_{i=0}^{m-2} a_{i+1} s_i$$

$$G_T(\underline{s}) = s_{m-1}\left[1 + \sum_{i=0}^{m-2} a_{i+1} s_i + \sum_{i=0}^{m-2} a_{i+1} s_i + a_0\right]$$

$$+ (a_0 + \sum_{i=0}^{m-2} a_{i+1} s_i)(\sum_{i=0}^{m-2} a_{i+1} s_i)$$

$$= \left[s_{m-1} + \sum_{i=0}^{m-2} a_{i+1} s_i\right](a_0 + 1)$$

$$\text{if } a_0 = 1 \qquad G_T^{(\underline{s})} = 0$$

$$a_0 = 0 \qquad G_T^{(\underline{s})} = s_{m-1} + \sum_{i=0}^{m-2} a_{i+1} s_i$$

The condition $G_T^{(\underline{s})} = 0$ can be used as a test for non-singularity of a feedback shift register. However there is another test.[4] This one has the advantage that if $G_T^{(\underline{s})} \neq 0$, then information as to the set of starting states is obtained.

The usefulness of an analytic expression for the set of starting states is not at all clear. However, knowledge of the set of starting states is important in several algorithms used for cycle structure analysis of nonlinear Feedback Shift Registers.

BIBLIOGRAPHY

1. J. Massey and R. Liu, "A New Diagram for Feedback Shift Registers," Proceedings, 3rd Annual Allerton Conference on Circuit and System Theory, University of Illinois, October, 1965, pp. 73-81.
2. F. Mowle and D. Sullivan, "Cycle Structure of Monotone Feedback Shift Registers," Proceedings, 3rd Annual Allerton Conference on Circuit and System Theory, University of Illinois, October, 1965, pp. 82-90.
3. J. Massey and R. Liu, "Monotone Feedback Shift Registers," Proceedings, 2nd Annual Allerton Conference on Circuit and System Theory, University of Illinois, September, 1964 pp. 860-874.
4. S. Golomb, Shift Register Sequences. Holden-Day, Inc., San Francisco, 1967.

REAL TIME SCALING*

A. PAPOULIS
Polytechnic Institute of Brooklyn
Brooklyn, New York

Abstract

The problem of determining a system yielding as output the signal f(ct) is considered. It is shown that such a system can be realized by two all-pass filters with quadratic phase and a linear-FM generator. The solution is motivated by the following optical analog: on its image plane, an object f(x) generates the field f(cx).

PRINCIPAL RESULT

Given a signal f(t) and a constant c , our objective is to find a system yielding as output the signal f(ct). In the course of the analysis, we shall also reexamine the problem of pulse compression and the method of designing a real time spectrum analyzer. As we mention at the end of the paper, all our results are direct analogies of corresponding optical phenomena.

We propose as solution of the time scaling problem the system of Fig. 1 consisting of two all-pass filters with linear group delay

$$H_1(\omega) = e^{j\pi/4} \sqrt{\frac{\pi}{\alpha}}\, e^{-j\omega^2/4\alpha} \qquad H_2(\omega) = e^{j\pi/4} \sqrt{\frac{\pi}{\gamma}}\, e^{-j\omega^2/4\gamma} \tag{1}$$

and a signal generator

$$s(t) = e^{-j\beta t^2} \tag{2}$$

The input to the first filter is the signal f(t). Its output r(t) multiplied by s(t) is the input a(t) of the second filter

$$a(t) = r(t)\, e^{-j\beta t^2} \tag{3}$$

We maintain that, if

$$\alpha + \gamma = \beta \tag{4}$$

---

* This work was supported wholly by the Joint Services Electronics Program (U. S. Army, U. S. Navy, U. S. Air Force) under Contract No. AF 49(638)-1402.

(lens formula), then the output g(t) of the second filter is given by

$$g(t) = \frac{\pi}{a} e^{j\gamma\beta t^2/a} f(-\gamma t/a) \tag{5}$$

If the duration of f(t) equals T

$$f(t) = 0 \qquad \text{for} \qquad |t| > T/2$$

and

$$a\beta T^2 << \gamma \tag{6}$$

then

$$e^{j\gamma\beta t^2/a} \simeq 1 \qquad \text{for} \qquad |\gamma t/a| < T$$

and (5) yields

$$g(t) \simeq \frac{\pi}{a} f(ct) \qquad c = -\gamma/a \tag{7}$$

ANALYSIS

From the Fourier transform pair

$$e^{jat^2} \longleftrightarrow e^{j\pi/4} \sqrt{\frac{\pi}{a}}\, e^{-j\omega^2/4a}$$

and (1), it follows that the impulse responses of the two filters are

$$h_1(t) = e^{jat^2} \qquad \text{and} \qquad h_2(t) = e^{j\gamma t^2} \tag{8}$$

respectively. Therefore, the output g(t) of the system of Fig. 1 is given by

$$g(t) = \left\{ \left[f(t) * e^{jat^2} \right] e^{-j\beta t^2} \right\} * e^{j\gamma t^2} \tag{9}$$

The quantity in the bracket is the output of the first filter

$$r(t) = \int_{-\infty}^{\infty} f(\tau)\, e^{ja(t-\tau)^2} d\tau = e^{jat^2} \int_{-\infty}^{\infty} f(\tau)\, e^{ja\tau^2} e^{-j2at\tau} d\tau \tag{10}$$

With $F_1(\omega)$ the Fourier transform of the function

$$f_1(t) = f(t)\, e^{jat^2} \tag{11}$$

(10) yields

$$r(t) = e^{jat^2} F_1(2at) \tag{12}$$

The quantity in braces in (9) is the input $a(t)$ to the second filter. Using (12) and (4), we thus conclude that

$$a(t) = e^{j\alpha t^2} F_1(2\alpha t)\, e^{-j\beta t^2} = e^{-j\gamma t^2} F_1(2\alpha t)$$

Finally, the output $g(t)$ is obtained by convolving the above with $e^{j\gamma t^2}$

$$g(t) = \int_{-\infty}^{\infty} e^{-j\gamma\tau^2} F_1(2\alpha\tau)\, e^{j\gamma(t-\tau)^2} d\tau = e^{j\gamma t^2} \int_{-\infty}^{\infty} F_1(2\alpha\tau)\, e^{-j\gamma t\tau}\, d\tau \tag{13}$$

From the Fourier inversion formula and (11) it follows that

$$g(t) = \frac{\pi}{\alpha} e^{j\gamma t^2} f_1(-\gamma t/\alpha) = \frac{\pi}{\alpha} e^{j\gamma t^2} e^{j\gamma^2 t^2/\alpha} f(-\gamma t/\alpha) \tag{14}$$

and (5) results.

Pulse generator.

In the preceding discussion we assumed that the generator output $s(t)$ is given by (2) for all t. We shall now assume that $s(t)$ equals zero for $|t| > a$

$$s(t) = e^{-j\beta t^2} p_a(t) \qquad p_a(t) = \begin{cases} 1 & |t| < a \\ 0 & |t| > a \end{cases} \tag{15}$$

In this case, the output $r(t)$ of the first filter is not changed and the input $a(t)$ to the second filter is given by [see (12) and (4)]

$$a(t) = r(t)\, s(t) = e^{j\gamma t^2} F_1(2\alpha t)\, p_a(t) \tag{16}$$

Hence, (13) still holds provided $F_1(2\alpha\tau)$ is replaced by $F_1(2\alpha\tau)p_a(\tau)$. To determine $g(t)$, it suffices, therefore, to replace in (14) the function $f_1(t)$ by the inverse Fourier transform $f_2(t)$ of the product

$$F_2(\omega) = F_1(\omega)\, p_a(\omega/2\alpha) = F_1(\omega)\, p_{2\alpha a}(\omega) \tag{17}$$

Since

$$f_1(t) \longleftrightarrow F_1(\omega) \qquad \frac{\sin 2\alpha a t}{\pi t} \longleftrightarrow p_{2\alpha a}(\omega)$$

it follows from the convolution theorem that

$$f_2(t) = f_1(t) * \frac{\sin 2\alpha at}{\pi t} = \left[f(t)\, e^{j\alpha t^2}\right] * \frac{\sin 2\alpha at}{\pi t} \tag{18}$$

This expression can be simplified with the help of the following:

Lemma. If $y(t) = 0$ for $|t| > e$ and

$$\alpha e^2 << 1 \tag{19}$$

then

$$\left[e^{j\alpha t^2} x(t)\right] * y(t) \simeq e^{j\alpha t^2} \left[x(t) * y(t)\right] \tag{20}$$

Proof. From (19) it follows that

$$e^{j\alpha(t-\tau)^2} \simeq e^{j\alpha t^2} \qquad \text{for} \qquad |\tau| < e$$

Since $y(t) = 0$ for $|t| > e$, the left side of (20) equals

$$\int_{-e}^{e} e^{j\alpha(t-\tau)^2} x(t-\tau)\, y(\tau)\, d\tau \simeq e^{j\alpha t^2} \int_{-e}^{e} x(t-\tau)\, y(\tau)\, d\tau$$

and (20) results.

We return to (18). The function $\sin 2\alpha at/\pi t$ is negligible for $|t| > \pi/\alpha a$. If, therefore,

$$\alpha a^2 >> \pi^2 \tag{21}$$

then [see (20)]

$$f_2(t) = e^{j\alpha t^2} \left[f(t) * \frac{\sin 2\alpha at}{\pi t}\right] \tag{22}$$

Thus: If $s(t)$ is a linear-FM pulse as in (15), then the output $g(t)$ of the system of Fig. 1 is given by

$$g(t) = \frac{\pi}{\alpha}\, e^{j\gamma\beta t^2/\alpha}\, \varphi(-\gamma t/\alpha) \tag{23}$$

where

$$\varphi(t) = f(t) * \frac{\sin 2\alpha at}{\pi t} \tag{24}$$

Spectral analysis and pulse compression.

Consider the system of Fig. 2 with input

$$f(t)\, e^{-j\beta t^2}$$

and impulse response $e^{j\beta t^2}$. This system results by eliminating the first filter in Fig. 1 and setting $\beta = \gamma$. The response $g(t)$ is given by

$$g(t) = \int_{-\infty}^{\infty} f(\tau)\, e^{-j\beta\tau^2}\, e^{j\beta(t-\tau)^2}\, d\tau = e^{j\beta t^2} \int_{-\infty}^{\infty} f(\tau)\, e^{-j2\beta t\tau}\, d\tau$$

With $F(\omega)$ the Fourier transform of $f(t)$, we thus have

$$g(t) = e^{j\beta t^2} F(2\beta t) \tag{25}$$

If $F(\omega) = 0$ for $|\omega| > \Omega$ and

$$\Omega^2 < < \beta \tag{26}$$

then (25) yields

$$g(t) \simeq F(2\beta t) \tag{27}$$

This known result shows that $g(t)$ equals the Fourier transform of $f(t)$ properly scaled. In other words, the system is a spectrum analyzer.

If the generator signal $s(t)$ is a pulse as in (15), then the input to the system equals

$$f(t)\, p_a(t)\, e^{-j\beta t^2}$$

hence, (25) still holds provided $F(\omega)$ is replaced by the Fourier transform

$$F_3(\omega) = F(\omega) * \frac{\sin a\omega}{\pi\omega} \tag{28}$$

of the signal $f(t)\, p_a(t)$.

Pulse Compression. Suppose now that $f(t) = 1$. In this case,

$$F_3(\omega) = \frac{2 \sin a\omega}{\pi\omega}$$

and (25) yields

$$g(t) = e^{j\beta t^2} \frac{\sin 2\beta a t}{\beta t} \tag{29}$$

If $\beta a^2 >> 1$, then the above reduces to

$$g(t) = \frac{\sin 2\beta a t}{\beta t} \tag{30}$$

Thus, a linear-FM pulse of duration 2a passing through a quadratic phase filter (Fig. 3) is compressed into the pulse (30) of effective duration $\pi/\beta a$.

Band-pass signals.

The preceding results can be readily extended to band-pass signals and filters. We shall assume that:

the input to our system equals

$$f(t)\, e^{j\omega_o t}$$

the impulse responses of the two filters are given by

$$e^{j\alpha t^2}\, e^{j\omega_1 t} \qquad\qquad e^{j\gamma t^2}\, e^{j\omega_3 t}$$

respectively, and the signal generator output by

$$e^{-j\beta t^2}\, e^{j\omega_2 t}$$

The system output is now given by

$$g(t) = \left\{ \left[f(t)\, e^{j\omega_o t} * e^{j\alpha t^2}\, e^{j\omega_1 t} \right] e^{-j\beta t^2}\, e^{j\omega_2 t} \right\} * \left[e^{j\gamma t^2}\, e^{j\omega_3 t} \right] \tag{31}$$

If

$$\alpha + \gamma = \beta \qquad \text{and} \qquad \omega_3 = \omega_1 + \omega_2$$

then (31) yields

$$g(t) = \frac{\pi}{\alpha}\, e^{j\gamma\beta t^2/\alpha}\, f(-\gamma t/\alpha)\, e^{-j(\omega_o - \omega_1)\gamma t/\alpha}\, e^{j\omega_3 t} \tag{32}$$

as in (5).

If $\omega_3 \neq \omega_1 + \omega_2$, then, except for a constant factor, the right side of (32) is multiplied by

$$e^{j\gamma t(\omega_3 - \omega_1 - \omega_2)/\alpha}$$

and in the term $f(-\gamma t/\alpha)$ the time t is replaced by $t - (\omega_1 + \omega_2 - \omega_3)/\gamma$.

The corresponding result for the spectrum analyzer follows from the equation

$$\left[f(t)\, e^{j\omega_o t}\, e^{j\beta t^2}\, e^{j\omega_2 t} \right] * \left[e^{j\beta t^2}\, e^{j\omega_3 t} \right]$$

$$= e^{j\beta t^2}\, e^{j\omega_3 t}\, F(-2\beta t + \omega_o + \omega_2 - \omega_3).$$

OPTICAL ANALOGIES

We conclude with a brief discussion of the corresponding optical phenomena.

Consider a monochromatic object

$$f(x)\, e^{-j\omega t} \qquad k = \omega/c = 2\pi/\lambda$$

on the $z = 0$ plane. The amplitude $r(x)$ of the resulting field on a plane P^- distance z_o from the object (Fig. 4) is given by

$$r(x) = \int_{-\infty}^{\infty} f(\xi)\, e^{j\alpha(x-\xi)^2}\, d\xi \qquad \alpha = k/2z_o \tag{33}$$

(within a factor). This results readily from Kirchhoff's formula and Fresnel's approximation. A lens of focal length f acts as a spatial filter with transmission function

$$e^{-j\beta x^2} \qquad \beta = k/2f$$

If, therefore, it is placed on the $z = z_o$ plane, then the resulting field on the plane P^+ just to the right of the lens will equal

$$a(x) = r(x)\, e^{-j\beta x^2} \tag{34}$$

Finally, the field $g(x)$ on the plane P_i distance z_1 from the lens equals

$$g(x) = a(x) * e^{j\gamma x^2} \qquad \gamma = k/2z_1 \tag{35}$$

as in (33). If z_1 is such that

$$\frac{1}{z_o} + \frac{1}{z_1} = \frac{1}{f} \tag{36}$$

then $\alpha + \gamma = \beta$ and (35) yields [see (5)]

$$|g(x)| = |f(-z_o x/z_1)| \tag{37}$$

Thus, on the image plane P_i (the intensity of) the image $g(x)$ is proportional to the object magnified by the factor z_1/z_o.

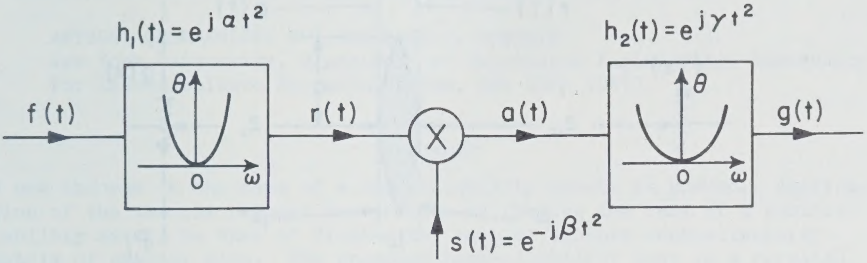
The optical analog of the assumption that the generator is of finite duration [see (15)] is the fact that the lens is of finite size. Hence, if the radius of the lens equals a, then $g(x)$ is similar not to $f(x)$, but to the signal

$$f(x) * \frac{\sin kax/z_o}{\pi x}$$

[see (23) and (33)].

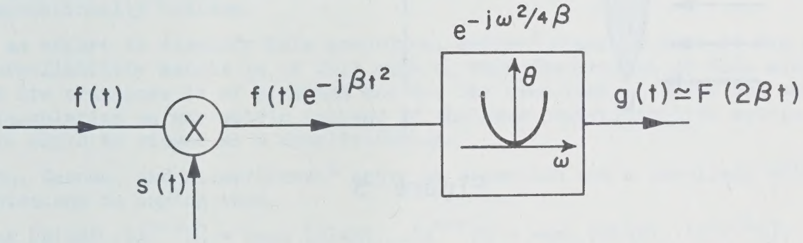
The spectrum analyzer formula (27) corresponds to the fact that the field on the focal plane of a lens (Fig. 5) is proportional to the Fourier transform of the incident wave.

Finally, the pulse compression principle has as optical equivalent the concentration of the incoming wave into a narrow region of the focal plane of the lens (Fig. 6).



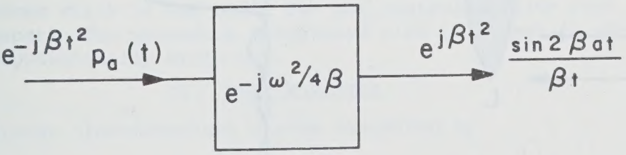
$h_1(t) = e^{j\alpha t^2}$
θ
0
ω
f(t)
r(t)
a(t)
$h_2(t) = e^{j\gamma t^2}$
θ
0
ω
g(t)
$s(t) = e^{-j\beta t^2}$

Figure 1



$e^{-j\omega^2/4\beta}$
f(t)
$f(t)\,e^{-j\beta t^2}$
θ
ω
$g(t) \simeq F(2\beta t)$
s(t)

Figure 2



$e^{-j\beta t^2} p_a(t)$
$e^{-j\omega^2/4\beta}$
$e^{j\beta t^2}$
$\frac{\sin 2\beta a t}{\beta t}$

Figure 3

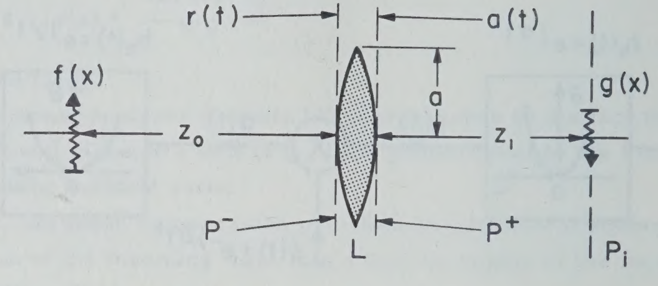

r(t)
a(t)
f(x)
g(x)
a
Z₀
Z₁
P⁻
P⁺
L
Pᵢ

Figure 4

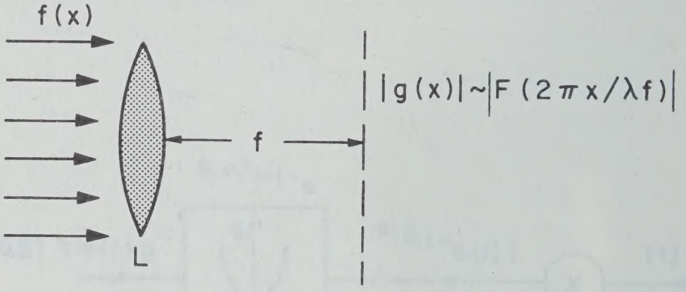

f(x)
|g(x)|~|F(2πx/λf)|
f
L

Figure 5

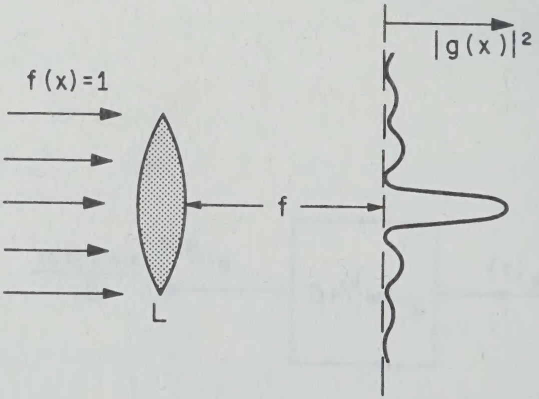


Figure 6

A NEW TEST FOR RANK OF A CONTROLLABILITY-OBSERVABILITY MATRIX

ANTHONY GRAMMATICOS and MARSHALL I. EPSTEIN
New York University, Department of Electrical Engineering, Laboratory for Electroscience Research, Bronx, New York 10453

ABSTRACT

A new theorem on the rank of a controllability matrix is proved. Application of the theorem reduces the problem of finding the rank of a controllability matrix to that of finding the rank of another controllability matrix of smaller size. The proposed controllability test is a repeated systematic application of this theorem. The procedure involved in carrying out the test is straightforward and computationally efficient.

I. INTRODUCTION

The concept of controllability of a dynamical system was introduced first by Kalman.[1] A detailed study of this concept appears in Ref. 4. In order to test the controllability of a Time-Invariant Linear System described by its state differential equations, (see (1) below) one must test whether the nxmn controllability matrix of the system (see (2) below) is of full rank n. In order to apply this test, given A and B, one must first calculate the controllability matrix and then use some standard method, such as triangularization, to determine its rank. Such a procedure is cumbersome and computationally tedious.

In an effort to simplify this procedure, Fowler[5] observed that if the nxmn controllability matrix is of full rank n, then the product of this matrix and its transpose is of size nxn and has the same rank n; hence one would triangularize an nxn matrix instead of the nxmn controllability matrix. This could be viewed as a simplification.

Chen, Desoer, and Niederlinski[6] prove an assertion and a corollary which is equivalent to saying that

$$\text{rank}\,[B\vdots AB\vdots..\vdots A^{n-1}B] = \text{rank}\,[B\vdots AB\vdots...\vdots A^{n-r}B] = \text{rank}\,[B\vdots AB\vdots..\vdots A^{n_o-1}B];$$

here, r is the rank of B and n_o is the degree of the minimal polynomial of A. Now, depending on the values of r and n_o, either one of the last two matrices may be of size considerably smaller than the size of the first matrix and hence the calculation of the rank will be simplified. The simplification based on the knowledge of r is considerable if $r>>1$; moreover r can be calculated easily. Based on the knowledge of n_o a simplification would result if $n_o<<n$; however this is not usually the case and furthermore it is not easy to calculate n_o.

In the present paper a new simplified controllability test for time-invariant linear systems is developed. The first part of the paper gives the proof of a theorem which is the basis for the controllability test discussed later in the paper. The procedure associated with this test is straightforward and computationally efficient.

II. PRELIMINARIES

Consider the linear time-invariant system described by

$$\dot{\underline{x}} = A\underline{x} + B\underline{u} \tag{1a}$$
$$\underline{y} = C\underline{x} + D\underline{u} \tag{1b}$$

This research was sponsored in part by the Air Force Office of Scientific Research, Office of Aerospace Research, U.S. Air Force, under AFOSR Grant AF-AFOSR-499-67.

where A,B,C, and D are constant nxn , nxm , pxn and pxm matrices respectively. Then, any state $\underline{x}$ is reachable from the origin in a finite time if and only if the rank of the nxmn matrix

$$[B \vdots AB \vdots \ldots\ldots \vdots A^{n-1}B] \tag{2}$$

is equal to n.[1,2]

III. A THEOREM

Let r>o be the rank of B and I_r the identity matrix of order r ; then there exist nonsingular matrices Q,R of size nxn and mxm respectively such that[3]

$$H = QBR \triangleq \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & I_r \end{array}\right] \tag{3}$$

Now a matrix G can be defined and partitioned as follows

$$G \triangleq QAQ^{-1} = \left[\begin{array}{c|c} A_1 & B_1 \\ \hline E & F \end{array}\right] \tag{4}$$

where $A_1 B_1 E_1$ and F are of size (n-r)x(n-r) , (n-r)xr , rx(n-r) and rxr respectively.

Theorem 1. If A,B,A_1,B_1, and r are the entities as defined above, then

$$\text{rank}\,[B \vdots AB \vdots \ldots \vdots A^{n-1}B] = r + \text{rank}\,[B_1 \vdots A_1B_1 \vdots \ldots \vdots A_1^{n-r-1}B_1] \tag{5}$$

Before proving Theorem 1, let us first prove the following

Lemma 1. If A,B,G,H are the matrices as defined above then

$$\text{rank}\,[B \vdots AB \vdots \ldots \vdots A^{n-1}B] = \text{rank}\,[H \vdots GH \vdots \ldots \vdots G^{n-1}H] \tag{6}$$

moreover if

$$H_1 \triangleq \left[\begin{array}{c} 0 \\ \hline I_r \end{array}\right]$$

then

$$\text{rank}\,[H \vdots GH \vdots \ldots \vdots G^{n-1}H] = \text{rank}\,[H_1 \vdots GH_1 \vdots \ldots \vdots G^{n-1}H_1] \tag{7}$$

Proof of Lemma 1. With Q, R nonsingular matrices of appropriate size,

$$\text{rank}\,[B \vdots AB \vdots \ldots \vdots A^{n-1}B] =$$

$$\text{rank}\,[QB \vdots QAB \vdots \ldots \vdots QA^{n-1}B] =$$

$$\text{rank}\,[QBR \vdots QABR \vdots \ldots \vdots QA^{n-1}BR] \quad .$$

Now

$$QBR = H$$

$$QABR = QAQ^{-1}QBR = GH$$

$$QA^{n-1}BR = Q\underbrace{A\ldots\ldots A}_{n-1}BR$$

$$= \underbrace{QAQ^{-1}QAQ^{-1}\ldots\ldots QAQ^{-1}}_{n-1}QBR = G^{n-1}H \quad .$$

Hence

$$\text{rank}\,[B \vdots AB \vdots \ldots \vdots A^{n-1}B] = \text{rank}\,[H \vdots GH \vdots \ldots \vdots G^{n-1}H] \quad .$$

Furthermore note that

$$H = [0 \vdots H_1]$$

and that for any k

$$G^kH = G^k[0 \vdots H_1] = [0 \vdots G^kH_1] \quad .$$

Hence

$$\begin{aligned}\operatorname{rank}\,[H \vdots GH \vdots \ldots \vdots G^{n-1}H] &= \operatorname{rank}\,[0 \vdots H_1 \vdots 0 \vdots GH_1 \vdots \ldots \vdots 0 \vdots G^{n-1}H_1] \\ &= \operatorname{rank}\,[H_1 \vdots GH_1 \vdots \ldots \vdots G^{n-1}H_1]\end{aligned}$$

and the proof of Lemma 1 is complete.

Proof of Theorem 1.

One can verify easily that

$$GH_1 = \begin{bmatrix} B_1 \\ \cdots \\ 0 \end{bmatrix} + H_1F$$

$$G^2H_1 = \begin{bmatrix} A_1B_1 \\ \cdots\cdots \\ 0 \end{bmatrix} + GH_1F + H_1EB_1 \tag{8}$$

Now use mathematical induction to prove that for $k \geq 2$

$$G^kH_1 = \begin{bmatrix} A_1^{k-1}B_1 \\ \cdots\cdots \\ 0 \end{bmatrix} + G^{k-1}H_1F + \sum_{\ell=2}^{k} G^{k-\ell}H_1EA_1^{\ell-2}B_1 \tag{9}$$

For $k=2$ equation (9) check with (8). Assume that (9) is valid for some $k>2$; then prove that (9) is valid for $k+1$ as follows:
Premultiply both sides of (9) by G to obtain

$$G^{k+1}H_1 = GG^kH_1 = G\begin{bmatrix} A_1^{k-1}B_1 \\ \cdots\cdots \\ 0 \end{bmatrix} + GG^{k-1}H_1F + G\sum_{\ell=2}^{k} G^{k-\ell}H_1EA_1^{\ell-2}B_1 \tag{10}$$

Note that

$$G\begin{bmatrix} A_1^{k-1}B_1 \\ \cdots\cdots \\ 0 \end{bmatrix} = \begin{bmatrix} A_1^kB_1 \\ \cdots\cdots \\ EA_1^{k-1}B_1 \end{bmatrix} = \begin{bmatrix} A_1^kB_1 \\ \cdots\cdots \\ 0 \end{bmatrix} + H_1EA_1^{k-1}B_1 \tag{11}$$

Introduce (11) into (10) to obtain

$$G^{k+1}H_1 = \begin{bmatrix} A_1^kB_1 \\ \cdots\cdots \\ 0 \end{bmatrix} + G^kH_1F + G\sum_{\ell=2}^{k} G^{k-\ell}H_1EA_1^{\ell-2}B_1 + H_1EA_1^{k-1}B_1$$

or

$$G^{k+1}H_1 = \begin{bmatrix} A_1^kB_1 \\ \cdots\cdots \\ 0 \end{bmatrix} + G^kH_1F + \sum_{\ell=2}^{k+1} G^{k-\ell+1}H_1EA_1^{\ell-2}B_1 \tag{12}$$

which is identical to (9). Q.E.D.
Returning to (9) observe that the columns of

$$G^{k-1}H_1F + \sum_{\ell=2}^{k} G^{k-\ell}H_1EA_1^{\ell-2}B_1$$

are linear combinations of the columns of G^sH_1 where $s<k$ and they do not contribute to the rank of $[H_1 \vdots GH_1 \vdots \ldots \vdots G^{n-1}H_1]$; hence

$$\text{rank}\,[H_1 \vdots GH_1 \vdots \ldots \vdots G^{n-1}H_1] = \text{rank}\begin{bmatrix} 0 & \vdots B_1 & \vdots A_1B_1 & \vdots & \cdots & \vdots A_1^{n-2}B_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ I_r & \vdots 0 & \vdots 0 & \vdots & & \vdots 0 \end{bmatrix}$$

$$= r + \text{rank}\,[B_1 \vdots A_1B_1 \vdots \ldots \vdots A_1^{n-2}B_1] \quad .$$

Now since A_1 is an (n-r)x(n-r) matrix there exist at most (n-r-1) linearly independent blocks $A_1^t B_1$ according to the Cayley-Hamilton theorem.

Hence

$$\text{rank}\,[B_1 \vdots A_1B_1 \vdots \ldots \vdots A_1^{n-r-1}B_1 \vdots \ldots \vdots A_1^{n-2}B_1] = \text{rank}\,[B_1 \vdots A_1B_1 \vdots \ldots \vdots A_1^{n-r-1}B_1]$$

and hence

$$\text{rank}\,[H_1 \vdots GH_1 \vdots \ldots \vdots G^{n-1}H_1] = r + \text{rank}\,[B_1 \vdots \ldots \vdots A_1^{n-r-1}B_1] \quad \text{Q.E.D.}$$

IV. THE TEST FOR CONTROLLABILITY

The test for controllability proposed here is based on a repeated application of Theorem 1 and Eq. (5). This yields

$$\text{rank}\,[B \vdots AB \vdots \ldots \vdots A^{n-1}B] = r + \sum_i r_i \qquad (i = 1,2,\ldots) \tag{13}$$

where r_i = rank B_i and B_i is obtained from A_{i-1} and B_{i-1} in the way that B_1 was obtained from A and B. The system (1) is completely controllable if and only if

$$r + \sum_i r_i = n \tag{14}$$

Now each application of Theorem 1 requires the determination of new matrices Q,R and Q^{-1} having the properties implied by (3) and (4). In order to avoid the usual difficulties associated with the determination of Q,R, and Q^{-1} the following alternative procedure is suggested.

1. Reduce B to the form

$$B' = \begin{bmatrix} 0 \vdots P \\ \cdots\cdots \\ 0 \vdots I_r \end{bmatrix} \tag{15}$$

   by using permutations of rows ONLY and elementary column operations.

2. Corresponding to each interchanging of two rows of B interchange the corresponding rows AND columns of A to finally get A' .

3. Observe that

$$\begin{bmatrix} I_{n-r} & \vdots -P \\ \cdots & \cdots \\ 0 & \vdots I_r \end{bmatrix} \begin{bmatrix} 0 & \vdots P \\ \cdots & \cdots \\ 0 & \vdots I_r \end{bmatrix} = \begin{bmatrix} 0 & \vdots 0 \\ \cdots & \cdots \\ 0 & \vdots I_r \end{bmatrix} \tag{16}$$

as required by (3); hence define

$$Q' \triangleq \begin{bmatrix} I_{n-r} & \vdots -P \\ \cdots & \cdots \\ & \vdots I_r \end{bmatrix} \tag{17}$$

whose inverse is

$$(Q')^{-1} = \begin{bmatrix} I_{n-r} & \vdots P \\ \cdots & \cdots \\ 0 & \vdots I_r \end{bmatrix} \tag{18}$$

since

$$Q'(Q')^{-1} = I_n$$

4. If

$$A' = \left[\begin{array}{c:c} M_1 & M_2 \\ \hdashline M_3 & M_4 \end{array}\right] \tag{19}$$

where M_4 is an rxr square matrix and the remaining blocks are of appropriate sizes, then

$$Q'A'(Q')^{-1} = \left[\begin{array}{c:c} M_1 - PM_3 & (M_1-PM_3)P + M_2 - PM_4 \\ \hdashline M_3 & M_3P + M_4 \end{array}\right] \tag{20}$$

Compare (20) and (4) to obtain

$$A_1 = M_1 - PM_3 \tag{21}$$

$$B_1 = (M_1-PM_3)P + M_2 - PM_4 \tag{22}$$

Now having found A_1 and B_1 one can apply the same procedure to obtain r_1 , A_2 , and B_2 ; then repeat until rank $B_i = 0$.

V. EXAMPLE

Let

$$A = \begin{bmatrix} 4 & 0 & 3 & 2 \\ 0 & 3 & 4 & 1 \\ 1 & 6 & 8 & 2 \\ 0 & 4 & 6 & 8 \end{bmatrix} \text{ and } B = \begin{bmatrix} 3 & 4 \\ 6 & 8 \\ 0 & 0 \\ 0 & 3 \end{bmatrix}$$

$$B \sim \begin{bmatrix} 0 & 0 \\ 6 & 8 \\ 3 & 4 \\ 0 & 3 \end{bmatrix} \sim B' = \begin{bmatrix} 0 & 0 \\ 2 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \therefore\ r = 2 \quad , P = \begin{bmatrix} 0 & 0 \\ 2 & 0 \end{bmatrix}$$

Only the first and third rows of B were permuted; hence

$$A \sim \begin{bmatrix} 1 & 6 & 8 & 2 \\ 0 & 3 & 4 & 1 \\ 4 & 0 & 3 & 2 \\ 0 & 4 & 6 & 8 \end{bmatrix} \sim A' = \left[\begin{array}{cc:cc} 8 & 6 & 1 & 2 \\ 4 & 3 & 0 & 1 \\ \hdashline 3 & 0 & 4 & 2 \\ 6 & 4 & 0 & 8 \end{array}\right]$$

Now

$$M_1 = \begin{bmatrix} 8 & 6 \\ 4 & 3 \end{bmatrix} , \quad M_2 = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}$$

$$M_3 = \begin{bmatrix} 3 & 0 \\ 6 & 4 \end{bmatrix} , \quad M_4 = \begin{bmatrix} 4 & 2 \\ 0 & 8 \end{bmatrix}$$

$$PM_3 = \begin{bmatrix} 0 & 0 \\ 6 & 0 \end{bmatrix} , \quad A_1 = M_1 - PM_3 = \begin{bmatrix} 8 & 6 \\ -2 & 3 \end{bmatrix}$$

$$PM_4 = \begin{bmatrix} 0 & 0 \\ 8 & 4 \end{bmatrix} , \quad (M_1-PM_3)\,P = \begin{bmatrix} 12 & 0 \\ 6 & 0 \end{bmatrix}$$

$$B_1 = (M_1 - PM_3)P + M_2 - PM_4 = \begin{bmatrix} 13 & 2 \\ -2 & -3 \end{bmatrix}$$ which is nonsingular, i.e. $r_1 = 2$.

Observe that

$$r + r_1 = 4 = n$$

hence the system is controllable.

VI. REMARKS ON THE AMOUNT OF COMPUTATION

At this point it is probably appropriate to compare the present method to the classical Gauss-Jordan method of triangularization.

The following table serves this purpose.

TABLE 1

m	n	Gauss-Jordan method	Present method
1	4	84	40
2	4	184	41
3	4	284	43
1	5	165	80
2	5	360	86
3	5	555	90
1	6	286	140
2	6	622	148
3	6	958	145
1	10	1330	660
2	10	2870	694
3	10	4410	719
5	10	7490	700
8	10	12110	647

Recall that m is the number of columns of B and n is the order of the system. The third column of Table 1 indicates estimates of the number of multiplications of elements of matrices to be executed if one is to start with given A (nxn) and B (nxm), calculate the controllability matrix, and find its rank by the Gauss-Jordan method of triangularization. The entries in the fourth column of Table 1 is the estimated number of multiplications of elements of matrices to be executed if one is to use the method proposed in the present paper; of course, starting again with given A and B and finding the rank of the controllability matrix.

The computational advantages of the proposed method over the classical Gauss-Jordan method of triangularization are obvious if one compares the corresponding entries in the third and fourth columns of Table 1.

The test also compares favorably with the simplified tests suggested by Fowler and Chen et al. Indeed, Fowler's procedure is hardly any simpler than the direct triangularization of the nxmn controllability matrix.

The simplified controllability test introduced by Chen, Desoer, and Niederlinski was described briefly in the introduction of this paper. There, it was pointed out that it is difficult to calculate n_o and that, even if n_o were found, the condition $n_o << n$ is an unusual case. Now with regard to the other simplification based on the knowledge of r and which was introduced by the same authors, it is interesting to note that one would obtain an analogous result if he were to apply the Theorem 1 only once and then find the rank of the resulting $(n-r) x r(n-r)$ matrix shown in the right-hand-side of (5); finding the rank of this matrix is easier than finding the rank of an $nxm(n-r)$ matrix as it is suggested in Ref. 6.

VII. CONCLUSIONS

A new controllability-observability test for continuous-and-discrete-time fixed linear systems has been developed. The approach taken was to develop a simple repetitive procedure which yields the rank of the controllability matrix. Such a method has definite computational advantages over the classical method of triangularization even if one were to apply triangularization to the simplified controllability tests that have been suggested by others more recently.

The condition for observability is analogous to the condition for controllability.[2] Hence one can replace B with C^T and A with A^T and follow exactly the same steps, as described above, to obtain the rank of the observability matrix. The conditions for controllability and observability of Discrete-time Fixed Linear Systems are the same with those of their continuous-time counterparts;[2] therefore the same tests can be applied.

REFERENCES

1. R.E. Kalman, "On the General Theory of Control," Proc. 1st IFAC Congress Butterworths, London, 1961.

2. L.A. Zadeh and C.A. Desoer, Linear System Theory, McGraw-Hill, New York 1963.

3. F.E. Hohn, Elementary Matrix Algebra, Macmillan, New York, 1958.

4. R.E. Kalman, Y.C. Ho, and K.S. Narendra, Controllability of Linear Dynamical Systems in "Contributions to Differential Equations," vol. 1, Interscience Publishers, Inc., New York, 1962.

5. W.T. Fowler, "A Simplified Controllability Test for Constant Linear Dynamical Systems," Trans. IEEE, vol. AC-11, p. 749, October 1966.

6. C.T. Chen, C.A. Desoer, and A. Niederlinski, "Simplified Conditions for Controllability of Linear Time-Invariant Systems," Trans. IEEE, vol. AC-10, pp. 613-14, July 1966.

INNER PRODUCT PROCEDURES FOR OBTAINING PROCESS REPRESENTATIONS

LEE.E. OSTRANDER
Systems Research Center
Case Institute of Technology
Cleveland, Ohio

GERALD H. COHEN
Department of Electrical Eng.
University of Rochester
Rochester, New York

ABSTRACT

A generalized description of a class of procedures for obtaining parameter models of a process from records of the process input and output is presented. This description is applicable to a large number of modeling procedures described in the literature. General techniques for comparing these alternative procedures and their performance are described using the theory of inner product vector spaces.

INTRODUCTION

An important step in the solution of many engineering problems is the determination of a mathematical description, or model, for the input-output relations of a process. Modeling procedures for accomplishing this step have been described by many investigators. However, all too often these procedures are isolated as separate entities and similarities of procedures are not considered. The absence of a general framework for modeling procedures has been noted, for instance, by Eykhoff[1] (p. 200).

The purpose of this paper is to describe a generalized formulation for a class of modeling procedures. The formulation makes use of inner product vector space concepts, and the modeling procedures are referred to as inner product procedures.

In keeping with a fairly common approach to description of modeling procedures the process is defined by a single input variable x(t) and a single output y(t). Such a representation preserves essential features of the modeling problem, and additional consideration for multi-input, multi-output systems can then be given separately.

THE MODEL

The model of the process is assumed to be of the generalized series form defined by Lubbock and Barker[3]:

$$\tilde{y}(t) = \sum_{k=1}^{N} \tilde{\alpha}_k h_k[x,t] \quad , \tag{1}$$

where $\tilde{y}(t)$ is the model output, and the set of model coefficients $\{\tilde{\alpha}_k\}$ specify the model for the individual process. The set of components $\{h_k[x,t]\}$ consist of known linear or nonlinear functions of time t and of the process input x(t). Examples of model components of this form are:

Linear Components

$$h_k[x,t] = \int_{-\infty}^{\infty} f_k(\tau)x(t-\tau)d\tau \quad , \tag{2}$$

$$h_k[x,t_\ell] = \sum_j f_k(t_j)x(t_\ell-t_j) \quad . \tag{3}$$

Nonlinear Components (used by Gabor, et al[5])

$$\tilde{y}(t) = \sum_i \tilde{\alpha}_i\, x(t-\tau_i) + \sum_i \sum_j \tilde{\alpha}_{ij}\, x(t-\tau_i)\, x(t-\tau_j)$$

$$+ \sum_i \sum_j \sum_k \tilde{\alpha}_{ijk}\, x(t-\tau_i)\, x(t-\tau_j)\, x(t-\tau_k) + \ldots \tag{4}$$

In equations (3) and (4) the subscripted t is used for the sampling instant in the sample form of the model components.

THE MODELING PROCEDURE

In the modeling problem under consideration, the model parameters $\tilde{\alpha}_k$ are determined from observations of the process input and output. The modeling procedure can thus be described by the equation

$$\underset{\rightarrow}{\tilde{\alpha}} = \underset{\rightarrow}{\Theta}[x,y,t] \quad , \tag{5}$$

where $\underset{\rightarrow}{\tilde{\alpha}}$ is the vector of model coefficients , and $\underset{\rightarrow}{\Theta}$ is the corresponding vector of functionals which maps the observed process input x(t) and output y(t) into the vector of coefficients. Choice of a modeling procedure is a choice of a vector of functionals. Choice of a model for a given process is a choice of model coefficients.

CHOICE OF FUNCTIONALS $\underset{\rightarrow}{\Theta}$

In the inner product procedures, the functionals $\underset{\rightarrow}{\Theta}$ are chosen to minimize a measure of error <e,e>, where e(t) is the error between the process and model outputs. The measure of error is an inner product functional with the following properties:[2] For any two functions a(t), b(t),

$$\begin{aligned} &\text{(i)} \quad < a,b > = <b,a > , \\ &\text{(ii)} \quad < \alpha a+\beta b,c > = \alpha < a,c > + \beta < b,c > , \\ &\text{(iii)} \quad < a,a > \geq 0 \end{aligned} \tag{6}$$

It should be noted that a possible restriction on the inner product space,

$$<a,a> = 0 \text{ only if } a(t) = 0, \tag{7}$$

is not required, and the absence of this restriction eliminates the need of separate definition of the vector space for each functional used.

Examples of inner product measures of error are the following,

$$\begin{aligned} <e,e>_1 &= \int_{t_\mu}^{t_\nu} w(t)\, e^2(t)\, dt \\ <e,e>_2 &= E[e^2(t_o)] \\ <e,e>_3 &= \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} e^2(t)\, dt \\ <e,e>_4 &= \int_{-\infty}^{\infty} \{w(t) * e(t)\}^2\, dt \\ <e,e>_5 &= \sum_{j=\mu}^{\nu} w(t_j)\, e^2(t_j) \end{aligned} \tag{8}$$

where w(t) is a weighting function, E denotes an ensemble average, and * denotes convolution.

The inner product can also be a function of the number or duration of the observations by defining the inner product error for an observation interval k as $<e,e>^{(k)}$. The finite average error can then be defined as

$$<e,e>_M = \frac{1}{J} \sum_{i=1}^{J} <e,e>^{(i)} \quad . \tag{9}$$

By introducing the operational definitions

$$< \alpha a,b >_M \triangleq \frac{1}{J} \sum_{i=1}^{J} < \alpha a,b >^{(i)} \quad , \tag{10}$$

$$< a+b,c >_M \triangleq \frac{1}{J} \sum_{i=1}^{J} < a+b,c >^{(i)} \quad , \tag{11}$$

the resulting function satisfies the inner product properties given in equation (6). If the estimating inner products $<e,e>^{(k)}_{est}$ are satistically independent for each of the observation intervals, $k = 1, \ldots, J$, then the ccefficient covariances decrease inversely with the number J of observation intervals.^{4}

Some general forms for the inner product functional can also be described. Error functionals $<e,e>_1$ and $<e,e>_4$ are special cases of the functional:

$$<e,e> \;=\; \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(s,t)\; e(s)\; e(t)\; dt\; ds \quad . \tag{12}$$

The functional $<e,e>_2$ is a apecial case of the functional

$$<e,e>= E \int_{t\mu}^{t\nu} \int_{t\mu}^{t\nu} \phi(s,t)=e(s)\; e(t)\; dt\; ds \quad . \tag{13}$$

The functional $<e,e>_5$ is a special case of the functional

$$<e,e> \;=\; \sum_{j=\mu}^{\nu} \sum_{k=\mu}^{\nu} \phi(s_j, t_\kappa)\; e(s_j)\; e(t_\kappa) \quad . \tag{14}$$

SOLUTIONS FOR MINIMUM ERROR

Let:

$$<e,\underset{\rightarrow}{h}>=\begin{bmatrix} <e,h_1> \\ <e,h_2> \\ \cdot \\ \cdot \\ \cdot \\ <e,h_N> \end{bmatrix};\ <y,\underset{\rightarrow}{h}>=\begin{bmatrix} <y,h_1> \\ <y,h_2> \\ \cdot \\ \cdot \\ \cdot \\ <y,h_N> \end{bmatrix}; \tag{15}$$

$$<\underset{\rightarrow}{h},\underset{\rightarrow}{h}>=\begin{bmatrix} <h_1,h_1><h_1,h_2> & \ldots <h_1,h_N> \\ <h_2,h_1><h_2,h_2> & \ldots <h_2,h_N> \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ <h_N,h_1><h_N,h_2> & \ldots <h_N,h_N> \end{bmatrix} \tag{16}$$

It is readily shown[4] that any inner product measure of error, satisfying equations (6), has a minimum when

$$<e,\underset{\rightarrow}{h}> =0, \tag{17}$$

or upon substitution for $e(t)=y(t)-\tilde{y}(t)$,

$$<y,\underset{\rightarrow}{h}> - \underset{\rightarrow}{\tilde{\alpha}}<\underset{\rightarrow}{h}\ ,\ \underset{\rightarrow}{h}> =0 \tag{18}$$

When the model components are linearly independent with respect to the given inner product, the matrix $<\underset{\rightarrow}{h},\underset{\rightarrow}{h}>$ is invertible and

$$\underset{\rightarrow}{\tilde{\alpha}} = (<\underset{\rightarrow}{h},\underset{\rightarrow}{h}>)^{-1}\ <y,\underset{\rightarrow}{h}> \tag{19}$$

ESTIMATION

In assessing the affects of random disturbances on the process, the process is further assumed to have the representation

$$y(t) = g[x,t]+ n(t)\ , \tag{20}$$

where g[x,t] describes the contribution of the input x(t) to the output y(t), and n(t) is the random disturbance of the process.

The "ideal" model coefficients $\underset{\rightarrow}{\hat{\alpha}}$ are defined as those coefficients which minimize a complete time or ensemble average of an inner product measure of error denoted $<e,e>_{ave}$, such as$<e,e>_2$ or $<e,e>_3$. However, a complete time or ensemble average functional cannot be implemented, since the ensemble or time interval is infinite. An estimate of model coefficients is obtained by minimizing an estimating inner product measure of error, denoted $<e,e>_{est}$.

Statistical evaluation of the estimates then proceeds by relating $<e,e>_{ave}$ and$<e,e>_{est}$. By introducing $\hat{e}$ (t),

$$\hat{e}(t) \triangleq g[x,t] - \sum_{k=1}^{N} \hat{\alpha}_k h_k[x,t] \quad , \tag{21}$$

equation (20) can be written as

$$y(t) = \sum_{k=1}^{N} \hat{\alpha}_k h_k[x,t] + \hat{e}(t) + n(t) \quad . \tag{22}$$

Subsequent substitution of equation(22) into equation(19)

$$\underset{\rightarrow}{\tilde{\alpha}} - \underset{\rightarrow}{\hat{\alpha}} = <\hat{e},\underset{\rightarrow}{h}> + <n,\underset{\rightarrow}{h}> \tag{23}$$

This last equation relates the error between the estimating and ideal coefficients to the noise term n(t) and the inherent model-process error which is given in equation (21). A measure of the estimating error is given by the second order moment matrix defined as

$$S(\tilde{\alpha},\hat{\alpha}) = E\left\{ (\underset{\rightarrow}{\tilde{\alpha}} - \underset{\rightarrow}{\hat{\alpha}}) \; (\underset{\rightarrow}{\tilde{\alpha}} - \underset{\rightarrow}{\hat{\alpha}})^T \right\} \tag{24}$$

where T denotes transpose. A further discussion of equation (24) can be found in reference 4 listed at the end of the paper.

THE CHOICE OF MODELING PROCEDURE

The inner product procedure is specified by choosing model components and an inner product functional or estimating functional. The functionals $\underset{\rightarrow}{\theta}$ result from the specification of the error criterion. The model components, on the other hand, have not yet been specified. It is relevent to describe the following method for comparing alternative choices of model components.

The error $<e,e>_{ave}$ may be written as

$$<e,e>_{ave} = \underset{\tilde{\alpha}}{\mathrm{Min.}} <e,e>_{ave} + \left[(\tilde{\vec{\alpha}} - \hat{\vec{\alpha}})^T < \vec{h},\vec{h}>_{ave} (\tilde{\vec{\alpha}} - \hat{\vec{\alpha}}) \right] \qquad (25)$$

where $\mathrm{Min.}<e,e>_{ave}$ is the value of $<e,e>_{ave}$ when the model coefficients $\tilde{\alpha}$ have their optimum values $\hat{\alpha}$. The last term on the right above is a loss function. The model components may be replaced by an orthonormal set (by using linear transformation), yielding:

$$\mathrm{Min.}<e,e>_{ave} = \{ <y,y>_{ave} - \sum_{k=1}^{N} \tilde{\alpha}_k^2 \} , \qquad (26)$$

$$<e,e>_{ave} = <y,y>_{ave} - \sum_{k=1}^{N} \hat{\alpha}_k^2 + \sum_{k=1}^{N} (\tilde{\alpha}_k - \hat{\alpha}_k)^2 . \qquad (27)$$

The average value of $<e,e>_{ave}$ obtained from the ensemble of estimates $\tilde{\vec{\alpha}}$ will be

$$\Gamma = E\{ <e,e>_{ave} \} = <y,y>_{ave} - \sum_{k=1}^{N} \hat{\alpha}_k^2 + \sum_{k=1}^{N} E\{(\tilde{\alpha}_k - \hat{\alpha}_k)^2\} \qquad (28)$$

At this point, one can ask what is the average contribution of each model component to Γ. The effect of the k^{th} component is to increase Γ by $E\{(\tilde{\alpha}_k - \hat{\alpha}_k)^2\}$ and to reduce Γ by $\hat{\alpha}_k^2$. In terms of the above measure Γ of procedure performance, the addition of a model component can only be justified if the square of the mean for the coefficient estimate exceeds its variance.

OTHER RELATED PROCEDURES

Obvious extensions can be made in the form of the model and the error functional which still permit use of a least some of the aforementioned techniques. For instance, the model may be an implicit or feedback model of the form:

$$\tilde{y}(t) = \sum_{k=1}^{N} \tilde{\alpha}_k h_k[x,t] + \sum_{\ell=1}^{N} \beta_\ell h_\ell[y,t] , \qquad (29)$$

or else the generalized series model may be derived from an arbitrary parameter model by a Taylor series expansion of the model equation. Additionally, the error $e(t)$ minimized in the functional $<e,e>$ may be chosen to be of the more general form:

$$e(t) = \gamma(x,y,t) - \tilde{y}(t), \tag{30}$$

where γ is a specified function of the process inputs and outputs.

CONCLUSIONS

The inner product procedures provide a framework for a great many modeling procedures and allow for comparisons and contrasts by relating estimates to ideal coefficients as in equation(23). The preceding discussion of the inner product procedures generally does not indicate an optimum choice of procedure, but rather indicates the degree of freedom which is possible in choosing a modeling procedure and in providing for its analysis.

REFERENCES

1. Eykhoff,P., "Process Parameter Estimation," Progress in Control Engineering, vol.2, Academic Press, New York, 1964, pp. 162-208.

2. Friedman, B., Principles and Techniques of Applied Mathematics, John Wiley and Sons, Inc., New York, 1956, Chapters 1 and 2.

3. Lubbock,J. and Barker,H. "Solution of the Identification Problem," IEEE Trans. on Applications and Industry, vol.83,No.72, May, 1964, pp. 166-173.

4. Ostrander, Lee E. The Use of Inner product Concepts in Obtaining Process Representations (Ph.D. Thesis) University of Rochester, 1966.

5. Gabor,D., Wilby,W., and Woodcock,R. "A Universal Nonlinear Filter, Predictor, and Simulator which Optimizes Itself by a Learning Process," Proceedings of the IEEE, vol.108B, No.40, July 1961, pp.422-438.

ON THE CONSTRUCTION OF STATE SPACES

JAMES A. RESH
Michigan State University
East Lansing, Michigan

Abstract

Recent research —most notably, the work of Zadeh described in reference 1 and 2—has provided theoretical Systemists with a powerful tool: Concise and extremely general characterizations of the notion of state descriptions a system. The present paper supplies a general construction which, in the case of causal systems, is shown to produce valid state descriptions. This establishes causality as the necessary and sufficient condition for the existence of state descriptions. The construction may prove to be a powerful tool for further investigation of state descriptions.

Introduction

Our starting point is the definition of an abstract oriented object-- the mathematical correlate of a physical device or subsystem with identified input and output terminals. It is simply a set $\mathcal{O}$ whose elements are pairs of time functions. The appearance of a pair $(\hat{u}, \hat{y})$ in $\mathcal{O}$ is interpreted by statements to the effect that an excitation described by $\hat{u}$ will produce in the object a response described by $\hat{y}$. Thus, $\hat{u}$ is called an admissible input and $\hat{y}$ is called an admissible output, or a possible response to $\hat{u}$. The collection of all admissible inputs (outputs) is denoted by $\mathcal{U}$ ($\mathcal{Y}$). The domain of each function in $\mathcal{U}$ and $\mathcal{Y}$ is the time line T (which, for simplicity of exposition, we take to be the extended real line). By making no assumptions about the ranges of these functions we are able to discuss arbitrary objects in the language of single-input-single output objects.

The work of Zadeh [1, 2], with minor modifications by the present author [3, 4], has essentially settled the question of what is to be meant by a state description of $\mathcal{O}$. That is to say, it provides a complete set of criteria for use in testing any given description to see whether or not it qualifies for the title "state description". There are two basic ingredients, of course: Identification of a set Σ_{t_o} to represent the set of possible states ("internal conditions") of the object at any initial time t_o; and description of the output from t_o to any later time t as a function of the initial state and of the input from t_o to t. The precise definition is set forth in the following paragraphs.

First, a bit of notation. If f is a function whose domain of definition is a set D, and if D' is a subset of D, then $f|D'$ designates the unique function whose domain is D' and whose value for each $d \in D'$ satisfies $(f|D')(d)=f(d)$. It is to be emphasized that $f|D'$ is a function in its own right, so if f and g are two different functions whose domains contain D' and which happen to be *equal on D'* then $f|D'$ is indistinguishable from $g|D'$. Now, let $\mathcal{J}$ denote the collection of all half-open time intervals of the form $(\cdot,\cdot]$. For a given abstract oriented object we introduce the symbols $\mathcal{U}_I$, $\mathcal{Y}_I$ and $\mathcal{O}_I$ as names for the sets $\{\hat{u}|I \mid \hat{u}\in\mathcal{U}\}$, $\{\hat{y}|I \mid \hat{y}\in\mathcal{Y}\}$ and $\{(\hat{u}|I, \hat{y}|I) \mid (\hat{u},\hat{y}) \in \mathcal{O}\}$, where I may be any interval in $\mathcal{J}$.

For later reference we reproduce here some standard definitions. The object $\mathcal{O}$ is call *determinate* if, for each $\hat{u}\in\mathcal{U}$, there is a unique $\hat{y}$ such that $(\hat{u},\hat{y}) \in \mathcal{O}$. Probabilistic objects are the most familiar example of indeterminate objects. A non-probabilistic example is the electrical one port called the "norator", whose port voltage and current are arbitrary unrelated time functions. Next, the object is said to be *nonanticipative* if, for any two admissible inputs $\hat{u}$ and $\hat{u}'$ satisfying $\hat{u}|(-\infty, t_o]=\hat{u}'|(-\infty, t_o]$ for some time t_o, there always exist pairs $(\hat{u},\hat{y})$ and $(\hat{u},\hat{y}')$ in D such that $\hat{y}|(-\infty, t_o]=\hat{y}'|(-\infty, t_o]$. Roughly speaking, a nonanticipative object is one whose present output is unaffected by future values of the input. The ideal predictor is an example of a determinate but anticipative object. Finally, an object is said to be *causal* if it is both determinate and nonanticipative. Most macroscopic physical models are causal in this sense.

Now. Let Σ be a function which assigns to each $t\in T$ a conjectured *state space* of $\mathcal{O}$ at time t. And let $\bar{A}$—formally described as subset of $\{(I, \sigma_o, u, y) \mid I=(t_o, t] \in \mathcal{J}, \sigma_o\in\Sigma(t_o), u\in\mathcal{U}_I, y\in\mathcal{Y}_I\}$—be a conjectured *input-output-state relation* of $\mathcal{O}$. The proposition $(I, \sigma_o, u, y) \in \bar{A}$ will be interpreted to mean that if the object starts in state σ_o at time t_o and is subsequently subjected to input u from t_o to t then it will respond by producing the output y from t_o to t. The criteria for validity of such a conjectured pair $(\Sigma, \bar{A})$ as a *state description* of $\mathcal{O}$ are the following four consistency conditions:

(M1) *First mutual-consistency condition.* For each interval $I=(t_o, t]$ in $\mathcal{J}$: The pair (u, y) is in $\mathcal{O}_I$ if and only if there is a state σ_o in $\Sigma(t_o)$ for which the tuplet (I, σ_o, u, y) is in $\bar{A}$.

(M2) *Second mutual-consistency condition.* The state space $\Sigma(-\infty)$ at $t=-\infty$ is a unit set.

(S1) First self-consistency condition. For each interval $I=(t_o, t] \in \mathcal{I}$, each state $\sigma_o \in \Sigma(t_o)$ and each input $u \in \mathcal{U}_I$: There exists at most one output y such that $(I, \sigma_o, u, y) \in \bar{A}$.

Let D_I be the collection of all state, input-pairs (σ_o, u) for which there does exist an output y as described in (S1). And define a function $\bar{A}_I : D_I \to Y_I$ by the formula $\bar{A}_I(\sigma_o, u)=y$. Then the fourth requirement is

(S 2) Second self-consistency condition. For each $I_o=(t_o, t] \in \mathcal{I}$, $(\sigma_o, u) \in D_{I_o}$, and $I_1=(t_1, t_2] \in \mathcal{I}$: There exists at least one state $\sigma_1 \in \Sigma(t_1)$ with the property that if $(\sigma_1, u') \in D_{I_1}$ then $(\sigma_1, u') \in D_{I_1}$ and

$$\bar{A}_{I_1}(\sigma_1, u')=\bar{A}_{I_o I_1}(\sigma_o, u\,u') \big| I_1$$

Various aspects of this definition are examined in some detail by the cited references. Space limitations forbid any further discussion here.

The Construction

Given an object $\mathcal{O}$, we turn now to the construction of a special pair (Σ^*, A^*). In the Appendix, this pair is shown to constitute a state description of $\mathcal{O}$ if $\mathcal{O}$ is both determinate and nonanticipative. Since these conditions were previously [4] known to be necessary for the existence of a state description, they will thus be established as necessary and sufficient.

The idea is this. To predict the behavior of the object for times later than t_o it is generally not enough simply to know the input for times later than t_o. One must know something about the input from $-\infty$ up to t_o as well. This "something", however, doesn't usually have to be complete knowledge of the specific input: If we had some way of dividing the pre-t_o input histories into classes in such a way that two histories in the same class (different classes) always had the same (different) effects on the post-t_o behavior of the object, then the "something" could be taken to be knowledge of which class the pre-t_o input was in. Furthermore, nothing less than this would suffice: If what we know about the pre-t_o input leaves its class membership ambiguous then, from the way we characterized the classes, it's clear that our knowledge is consistent with at least two *distinct* future behaviors of the object. The remainder of this section is devoted to formulating this idea in terms sufficiently clear to admit a rigorous proof of its validity.

First, the construction of Σ^*. For each $t_o \in T$ and $u \in \mathcal{U}_{(-\infty, t_o]}$ let $\mathcal{H}_{t_o}[u]$ denote the set of all $v \in \mathcal{U}_{(-\infty, t_o]}$ which are *equivalent* to u in the following sense. For any $w \in \mathcal{U}_{(t_o, \infty]}$ the concatenation uw is an admissible input if and only if vw is; and, in the cases for which both uw and vw are admissible, there corresponds to each admissible output $\hat{y}$ which is a possible response to *one* of the two inputs, uw or vw, a possible response $\hat{z}$ to the other such that $\hat{y}|(t_o, \infty] = \hat{z}|(t_o, \infty]$. Thus, we are calling u and v equivalent if and only if they are indistinguishable by experiments performed on $\mathcal{O}$ *after* time t_o. That the relation so described is, indeed, an equivalence relation on $\mathcal{U}_{(-\infty, t_o]}$ is easily established, so we omit the proof. The collection $H_{t_o} \triangleq \{\mathcal{H}_{t_o}[u] \mid u \in \mathcal{U}_{(-\infty, t_o]}\}$ is thus a partition of $\mathcal{U}_{(-\infty, t_o]}$ into mutually exclusive, collectively inclusive subsets called *equivalence classes*.

We define Σ^* by taking its value $\Sigma^*(t_o)$ at the arbitrary time t_o to be any set of the same cardinality as the collection H_{t_o} of equivalence classes defined above, i.e. any set whose elements can be placed in 1-1 correspondence with the elements of H_{t_o}. And, to accomodate the second mutual-consistency condition, we take for $\Sigma^*(-\infty)$ any unit class (any set with exactly one element).

Second, the construction of $\bar{A}^*$. Let $I=(t_o, t]$ be an arbitrary interval in $\mathcal{J}$. We must describe which tuplets (I, σ_o, u, y) are in $\bar{A}^*$ and which aren't. To do this, we first select a 1-1 correspondence between the sets in H_{t_o} and the states in $\Sigma(t_o)$; then describe it by means of a function $C_{t_o} : \Sigma^*(t_o) \xrightarrow[\text{onto}]{1-1} H_{t_o}$. Note that, for each state $\sigma_o \in \Sigma^*(t_o)$, the entity $C_{t_o}(\sigma_o)$ is a particular equivalence class of elements in $\mathcal{U}_{(-\infty, t_o]}$. In the present general treatment there are no criteria to suggest selecting any one such correspondence over any other, so any choice will do. The important point is that our prescription of $\Sigma^*(t_o)$ guarantees the existence of a nonempty class of correspondences to choose from.

The tuplets (I, σ_o, u, y) in $\bar{A}^*$ with $t_o > -\infty$ are to be precisely those for which there exists a pair $(u_o, y_o) \in \mathcal{O}_{(-\infty, t]}$ such that

$u_o \mid (-\infty, t_o]$ is in the equivalence class $C_{t_o}(\sigma_o)$, $u_o \mid I$ equals u, and $y_o \mid I$ equals y. For $t_o = -\infty$, σ_o is to be the single element of $\Sigma^*(-\infty)$; and (u, y) is to vary over all of $\mathcal{O}_I$.

To clarify the above definitions of Σ^* and $\bar{A}^*$ suppose that at time t_o the object is in state σ_o and that input $u \in \mathcal{U}_{(t_o, t_1]}$ is applied. How do we compute the resulting output y from t_o to t_1 and the final state σ_1 at time t_1? Well, to say that $\mathcal{O}$ is in state σ_o at time t_o is to say that the input from time $-\infty$ up to time t_o was one or another of the functions in $C_{t_o}(\sigma_o)$. Let u_o be any such function. Then y is the t_o to t_1 portion of the object's response to $u_o u$. And σ_1 is the state which corresponds to the equivalence class $\mathcal{H}_{t_1}[u_o u]$, i.e. $\sigma_1 = C_{t_1}^{-1}(\mathcal{H}_{t_1}[u_o u])$.

As mentioned earlier, the proof that (Σ^*, $\bar{A}^*$) constitutes a state description of $\mathcal{O}$ (if $\mathcal{O}$ is determinate and nonanticipative) is deferred to the Appendix.

An Example

In this section we use the ideas presented above to determine the state space of an object $\mathcal{O}$ whose input, output-pairs $(\hat{u}, \hat{y})$ are those real-valued functions related by

$$\hat{y}(t) = \int_{-\infty}^{t} \hat{u}(x)\, \epsilon^{(t-x)^2} dx.$$

We assume $\hat{u}(t)$ to be 0 for all sufficiently large negative t and piecewise continuous.

As a first step in the construction of the state space of $\mathcal{O}$ at time t_o we must determine the equivalence class $\mathcal{H}_{t_o}[u]$ for an arbitrary function u in $\mathcal{U}_{(-\infty, t_o]}$. But the condition for $v \in \mathcal{H}_{t_o}[u]$ in the present case is

$$\int_{-\infty}^{t} (uw)(x)\, \epsilon^{(t-x)^2} dx = \int_{-\infty}^{t} (vw)(x)^{(t-x)^2} dx \ \forall\, w \in U_{(t, \infty]}.$$

This easily reduces to the requirement

$$\int_{-\infty}^{t_o} (u-v)(x)\, \epsilon^{(t-x)^2} dx = o \ \forall\, t \in (t_o, \infty],$$

which implies u=v. Thus the equivalence class $\mathcal{H}_{t_o}[u]$ for each $u \in \mathcal{U}_{(-\infty, t_o]}$ contains u as its only element.

The function Σ^* whose value for each $t_o \in (-\infty, \infty]$ is $\Sigma^*(t_o) = \mathcal{U}_{(-\infty, o]}$ provides a minimal time-invariant state space for the object under consideration. A natural 1-1 correspondence between $\Sigma^*(t_o)$ and H_{t_o} is one which associates $\mathcal{H}_{t_o}[u] \in H_{t_o}$ with $u' \in \Sigma^*(t_o)$, where u' is the function defined for $t \in (-\infty, 0]$ by $u'(t) = u(t - t_o)$.

Perhaps the most notable feature of the present obeect is the enormous size of its state space: the set of all real-valued functions defined and piecewise continuous on $[-\infty, 0]$ and equal to zero on some half-open interval including $-\infty$. In fact, the state of $\mathcal{O}$ at each moment reflects every detail of the past input: $\mathcal{O}$ is a continuous-time object with complete memory. This type of object deserves further study.

Conclusion

This paper provides a completion of the foundations of the Zadeh theory of state descriptions. The result should prove to be of considerable value in further study of the theory and its applications. It is, for instance, already being explored as a tool for the study of distributed networks.

References

[1] L. A. Zadeh and C. A. Desoer, Linear System Theory, McGraw-Hill, New York; 1963

[2] L. A. Zadeh, "The Concept of State in System Theory", Views on General System Theory (M. D. Mesarovic, ed.), Ch3, Wiley, New York; 1964

[3] J. A. Resh, "An Improvement in the State Axioms", CSL Technical Report R-282, University of Illinois, Urbana; March, 1966.

[4] J. A. Resh, "Improvements in the State Axioms", Conference Record, Tenth Midwest Symposium on Circuit Theory, Purdue University, Lafayatte, Indiana, May, 1967.

Appendix

We turn now to the proof that the pair $(\Sigma^* = \bar{A}^*)$ defined by our construction meets the requirements for state descriptions if $\mathcal{O}$ is causal. This is accomplished by verifying, in turn, each of the four consistency conditions. By placing this work in the Appendix we hope to minimize the required fervor of our apologies for the extreme terseness of the arguments.

Proof of "only if" portion of (Ml). What we must show is that $(u, y) \in \mathcal{O}_{(t_o, t]} \Rightarrow \exists\ \sigma_o \in \Sigma^*(t_o) \ni \big((t_o, t], \sigma_o, u, y\big) \in \bar{A}^*$. But $(u, y) \in \mathcal{O}_{(t_o, t]} \Rightarrow \exists\ (\hat{u}, \hat{y}) \in \mathcal{O} \ni (u, y) = (\hat{u} \,|\, (t_o, t], \hat{y} \,|\, (t_o, t])$. The state $\sigma_o = C_{t_o}^{-1}\Big(\mathcal{H}_{t_o}[u \,|\, (-\infty, t_o]]\Big)$ has the required property.

Proof of "if" portion of (Ml). To be shown is that $\big((t_o, t], \sigma_o, u, y\big) \in \bar{A}^* \Rightarrow (u, y) \in \mathcal{O}_{(t_o, t]}$. If $t_o = -\infty$ this follows immediately from the definition of A^*. Otherwise, $\exists\ (u_o, y_o) \in \mathcal{O}_{(-\infty, t]} \ni u = u_o \,|\, (t_o, t]$

and $y = y_o \mid (t_o, t]$; but $(u_o, y_o) \in \mathcal{O}_{(-\infty, t]} \Rightarrow \exists\, (\hat{u}_o, \hat{y}_o) \in \mathcal{O} \ni u_o = \hat{u}_o \mid (-\infty, t]$, $y_o = \hat{y}_o \mid (t_o, t]$; so $u = \hat{u}_o \mid (t_o, t]$ and $y = \hat{y}_o \mid t_o, t]$ and the definition of $\mathcal{O}(t_o, t]$ is seen to require that it contain (u, y).

The proof of (M2) is an immediate consequence of the definition of $(\Sigma^*, \bar{A}^*)$.

Proof of (S1). We must show that $\big((t_o, t], \sigma_o, u, y\big) \in \bar{A}^*$ and $\big((t_o, t], \sigma_o, u, y'\big) \in \bar{A}^* \Rightarrow y = y'$. For $t_o = -\infty$ this follows immediately from the definition of $(\Sigma^*, \bar{A}^*)$ and the causality of $\mathcal{O}$. For $t_o > -\infty$ we have: $\big((t_o, t], \sigma_o, u, y\big) \in \bar{A}^* \Rightarrow \exists\, (\hat{u}, \hat{y}) \in \mathcal{O} \ni \hat{u} \mid (-\infty, t_o] \in C_{t_o}(\sigma_o)$ and $(\hat{u} \mid t_o, t], \hat{y} \mid (t_o, t]) = (u, y)$; similarly, $\exists\, (\hat{u}', \hat{y}') \in \mathcal{O} \ni \hat{u}' \mid (-\infty, t_o] \in C_{t_o}(\sigma_o)$ and $(\hat{u}' \mid (t_o, t], \hat{y}' \mid (t_o, t]) = (u, y')$. Consider now the two concatenations $\hat{w} = (\hat{u} \mid -\infty, t_o]\, u (\hat{u} \mid (t, \infty]$ and $\hat{w}' = \hat{u}' \mid (-\infty, t_o]\, (u \mid (t, \infty])$. Both are members of $\mathcal{U}$; $\hat{w}$ is, because it equals $\hat{u}$; and $\hat{w}'$ is, because $\hat{u}' \mid (-\infty, t_o]$ is in the same equivalence class $C_{t_o}(\sigma_o)$ as $\hat{u} \mid (-\infty, t_o]$. By the determinateness of $\mathcal{O}$, $\hat{w}$ and $\hat{w}'$ determine unique pairs $(\hat{w}, \hat{z}')$, in $\mathcal{O}$. The proof is completed by combining the observations: $\hat{z} \mid (t_o, \infty] = z' \mid (t_o, \infty]$ (a consequence of the equivalence of $\hat{u} \mid (-\infty, t_o]$ and $\hat{u}' \mid (-\infty, t_o]$ and equality of $\hat{w} \mid (t_o, \infty]$ and $\hat{w}' \mid (t_o, \infty]$; $\hat{z} \mid (-\infty, t] = \hat{y} \mid (-\infty, t]$ (causality of $\mathcal{O}$); and $\hat{z}' \mid (-\infty, t] = \hat{y}' \mid (-\infty, t]$ (Causality again); $y = \hat{y} \mid (t_o, t]$ (definition of $\hat{y}$); and $y' = \hat{y}' \mid (t_o, t]$ (definition of $\hat{y}'$).

Proof of (S2). Refer to the Introduction for a statement of the condition. Observe that the conditions for (σ_o, u) to be in $D_{(t_o, t]}$ are simply that σ_o be in $\Sigma^*(t_o)$ and that $u_o u$ be in $\mathcal{U}_{(-\infty, t]}$ for some (hence every) member u_o of $C_{t_o}(\sigma_o)$. Take $\sigma_1 = C_{t_1}^{-1}(\mathcal{H}_{t_1}[u_o u])$. We will show that this σ_1 has the required properties. First suppose that $(\sigma_o, uu') \in D_{(t_o, t_2]}$; then $u_o uu' \in \mathcal{U}_{(-\infty, t_2]}\ \forall\, u_o \in C_{t_o}(\sigma_o)$; so $u'_o u' \in \mathcal{U}_{(-\infty, t_2]}\ \forall\, u'_o \in H_{t_1}[u_o u]$; so $(\sigma_1, u') \in D_{(t_1, t_2]}$. Second, note that by the definition of $\bar{A}^*$, $\bar{A}^*_{(t_1, t_2]}(\sigma_1, u')$ is the $(t_1, t_2]$ segment of a response to $(u_o u)u'$; and $\bar{A}^*_{(t_o, t_2]}(\sigma_1, u')$ is the $(t_o, t_2]$ segment of a response to $u_o(uu')$. But $u_o(uu') = (u_o u)u'$ and $\mathcal{O}$ is causal so these two responses are equal; so $\bar{A}^*_{(t_1, t_2]}(\sigma_1, u') = \bar{A}^*_{(t_o, t_2]}(\sigma_o uu') \mid (t_1, t_2]$.

Q. E. D.

TIME-SHARING ACTIVE RC FILTERS

THOMAS G. MANCUSO AND SAMUEL C. LEE
Department of Electrical Engineering, New York University, Bronx,
New York, 10453

ABSTRACT

A class of active RC time-varying networks of which the time-varying elements are changing at a rate much slower as compared to the applied signal frequencies has been studied. It is shown that by properly controlling the element variations, these networks can be used as time-sharing filters which may have applications in n-channel systems. Examples of such filters are given.

I. INTRODUCTION

Time-varying networks have been studied by a number of authors[1-6] in the past decade. Most recent work done in this area has been on the study of the stability of time-varying networks.[3-6] Because of the complexity of the problem, even for the simple case of two-element-kind networks with one or two time-varying elements, few attempts have been made to investigate their design or applications.[7] In this paper, a class of time-varying networks will be considered, with some elements which are periodically time-varying, with the frequencies much smaller in comparison to those of the applied signals. The network function for this class of time-varying networks can be approximated by rational functions of s, with the poles and zeros being functions of time, i.e.

$$F(s,t) \cong H(t)\ \frac{[s+z_1(t)]\,[s+z_2(t)]\,\ldots[s+z_m(t)]}{[s+p_1(t)]\,[s+p_2(t)]\,\ldots[s+p_n(t)]} \quad .$$

This class of time-varying networks will be referred to as S-time-varying networks.

The advent of economical and small solid state active elements has led to the practical design of active RC circuits. Other advances include the introduction of resistors and capacitors with variable characteristics and no moving parts, which can be used as the time-varying elements. Because of these late developments, in this paper, we shall study the general properties of S-time-varying networks and their applications - time-sharing filters.

In the active RC synthesis of an nth order transmission function, it has invariably been found impractical to design higher than second-order networks at a time. It has become common practice in active filter design to realize any given nth order transmission function by a cascade of first and second order networks.

Consider the n-channel system shown in Fig. 1a. Assume that one input and output is used at a time. An equivalent system may be obtained by properly arranging a switching system at the input and output. We shall define this system as an n-channel time-sharing system. Furthermore, if the n-channel system contains n filters (as shown in Fig. 1b), we can replace the n filters by a single filter that can take on the n filter characteristics of the n filters at n specified times. This filter is defined as a time-sharing filter. This is shown in Fig. 1c.

II. SINGLE-LOOP FEEDBACK S-TIME-VARYING NETWORKS

The simplest method of obtaining second order transfer functions is to use single-loop feedback circuits. These circuits are characterized by a single-forward gain stage and a passive feedback path, as shown in Fig. 2. Single-loop feedback circuits can be classified into two basic types. These are the (1) finite gain controlled source (2) infinite gain controlled source.

First, consider the finite gain controlled source circuits.[8] In order to achieve a complex pole-pair, the general transfer functions for this class of networks can be classified by the forms in the first column of (1).

$$\begin{aligned}
G_1(s,t) &= \frac{N_1(s,t)}{(b_2+Ka_2)s^2 + b_1 s + b_0}, & G_1'(s,t) &= \frac{N_1'(s,t)}{(1+K)s^2 + 3\omega_0 s + \omega_0{}^2} \\
G_2(s,t) &= \frac{N_2(s,t)}{b_2 s^2 + (b_1-Ka_1)s + b_0}, & G_2'(s,t) &= \frac{N_2'(s,t)}{s^2 + (3-K)\omega_0 s + \omega_0{}^2} \\
G_3(s,t) &= \frac{N_3(s,t)}{b_2 s^2 + b_1 s + (b_0+Ka_0)}, & G_3'(s,t) &= \frac{N_3'(s,t)}{s^2 + 3\omega_0 s + (1+K)\omega_0{}^2}
\end{aligned} \tag{1}$$

where K is the controlled source gain which is a function of time, N_i are numerator polynomials, and a_i and b_i are functions of the passive elements, i.e., RC products. The a_i and b_i are functions of time. In order to completely control the root loci of the complex pole pair, two degrees of freedom are required. By properly combining several of the variables, two convenient degrees of freedom can be obtained. For the three forms in the first column of Eq. (1), one can choose (1) the controlled source gain K and (2) the RC product, obtained by letting all the R's be equal and all C's equal. Defining $\omega_0(t) = 1/RC(t)$, the transfer functions become those shown in the second column of Eq. (1). It can be shown by letting $x = \sigma$ and $y = j\omega$, the equations of the root loci of the poles of the transfer functions in Eq. (1) are tabulated in Table 1, and the plots of the root loci are shown

Table 1 Root Loci Equations

Transfer function	For ω_0 constant	For K constant	Q
$G_1'(s,t)$	$\left(x + \frac{\omega_0}{3}\right)^2 + y = \frac{\omega_0{}^2}{9}$	$y = \pm\frac{\sqrt{4K-5}}{3}x$	$\frac{\sqrt{1+K}}{3}$
$G_2'(s,t)$	$x^2 + y^2 = \omega_0{}^2$	$y = \pm\frac{\sqrt{4-(3-K)^2}}{3-K}x$	$\frac{1}{3-K}$
$G_3'(s,t)$	$x = -\frac{3\omega_0}{2}$	$y = \pm\frac{\sqrt{4K-5}}{3}x$	$\frac{\sqrt{1+K}}{3}$

where $x = \sigma$ and $y = j\omega$

in Fig. 3. Note that from the loci plots one can see that $G_2'(s,t)$ is potentially unstable while $G_1'(s,t)$ and $G_3'(s,t)$ are absolutely stable.

To illustrate the finite gain controlled source circuit as a S-time-varying network, a Sallen and Key circuit will be used. See Fig. 4a. With $R_1(t) = R_2(t) = R(t)$; $C_1(t) = C_2(t) = C(t)$ and $\omega_0(t) = 1/RC(t)$, then the transfer function for this circuit is given by Eq. (2).

$$\frac{E_{out}}{E_{in}}(s,t) = \frac{K(t)[\omega_0(t)]^2}{s^2 + [3-K(t)]\omega_0(t)s + [\omega_0(t)]^2} \tag{2}$$

With $K(t)$ fixed at a constant value, the poles will move as shown in Fig. 4b, radially, with an angle θ. K is determined from a specified θ by $K=3-2\cos\theta$. The distance of the pole from the origin is equal to the value of $\omega_o(t)$.

The infinite gain single-loop feedback circuit can also be used as an S-time varying network. A lowpass example is given in Fig. 5a. By letting $R_1(t) = R_3(t) = R_4(t) = R(t)$, $C_2(t) = C(t)$ and $C_5(t) = C(t)/\gamma(t)$, the number of degrees of freedom are reduced from five to two, i.e. $\omega_o(t) = 1/RC(t)$ and $\gamma(t) = C_2(t)/C_5(t)$. The transfer function for this circuit is given by Eq. (3).

$$\frac{E_{out}}{E_{in}}(s,t) = \frac{-\gamma(t)[\omega_o(t)]^2}{s^2 + 3\omega_o(t)s + \gamma(t)[\omega_o(t)]^2} \tag{3}$$

The root locus for this example is given in Fig. 5b. However, in general, it is found that there is no systematic procedure for obtaining simplified equations for the infinite gain circuits. An example of a four-pole lowpass time-sharing filter using single-loop S-time-varying networks will be considered. This time-sharing filter is characterized by the pole-zero diagram in Fig. 6a and will be part of the time-sharing system shown in Fig. 6b. The transfer characteristics, as a function of time and switch position are shown in Fig. 6c. The realization of this filter will be accomplished by the cascade of two networks of the form in Fig. 4a, with stages 1 and 2 having $K_1 = 1.15$ for $\theta = 22.5^\circ$ and $K_2 = 2.235$ for $\theta = 67.5^\circ$. For both stages, $\omega_{o1}(t) = \omega_{o2}(t) = 1/RC(t)$.

It is well known that for operational amplifiers that the gain-bandwidth product is a constant. Thus to achieve a large bandwidth, the gain must be kept low. However, referring to Table 1, column 3, it is seen that for the $G_1(s,t)$ and $G_3(s,t)$ forms, to realize high-Q complex poles a high gain is required, e.g. for a Q of 10, K must be 900. For the $G_2(s,t)$ form, although low gain will realize high-Q poles, the sensitivity of the function in the neighborhood of the center frequency is high, e. g. for the Q of 10, the sensitivity is 30. As a consequence, using the finite gain controlled source S-time varying networks to realize high-Q poles, there will be a problem of either high gain requirements or high sensitivity. For the infinite gain circuit, in order to achieve high-Q poles, a large spread in element values is required. In order to overcome these short-comings of the single-loop circuits, another configuration shall be investigated.

III. MULTI-LOOP FEEDBACK S-TIME-VARYING NETWORKS

In order to eliminate some of the problems which have been shown in the single-loop feedback circuits, the feedback loop has been changed by the addition of active elements. This type of circuit is shown in Fig. 7. The transfer function is give by Eq. (4).

$$G(s,t) = -T_A(s,t)\frac{\mu(t)}{1 + \mu(t)A_V(s,t)} \tag{4}$$

where T_A is a passive network for determining the desired numerator polynomial and $A_V(s,t)$ is a second order transfer function given by Eq. (5).

$$A_V(s,t) = H(t)\frac{s^2 + a_1(t)s + a_o(t)}{s^2 + b_1(t)s + b_o(t)} = H(t)\frac{N(s,t)}{D(s,t)} \tag{5}$$

The forms of A_V given in Table 2, are used to obtain complex poles. The terms a_i and b_i in A_V will be functions of the elements in the form of RC products. The circuits chosen for A_V will in general contain two R's and two C's. By letting $R_1(t)=R_2(t)=R(t)$, $C_1(t)=C_2(t)=C(t)$ and $\omega_o(t)=1/RC(t)$,

Table 2

$$A_{V1}(s,t) = H(t)\,\frac{a_0}{D(s,t)} \qquad G(s,t) = -T_A(s,t)\frac{-\mu(t)(s^2+b_1 s+b_0)}{s^2+b_1 s+(b_0+\mu H a_0)}$$

$$A_{V2}(s,t) = H(t)\,\frac{s^2}{D(s,t)} \qquad G(s,t) = -T_A(s,t)\frac{-\mu(t)(s^2+b_1 s+b_0)}{(1+\mu H)s^2+b_1 s+b_0}$$

$$A_{V3}(s,t) = H(t)\,\frac{(s^2+a_0)}{D(s,t)} \qquad G(s,t) = -T_A(s,t)\frac{-\mu(t)(s^2+b_1 s+b_0)}{(1+\mu H)s^2+b_1 s+(b_0+\mu H a_0)}$$

Note: a_i and b_i are functions of time.

one degree of freedom is established. The second degree of freedom will be the loop gain term, μH. The exact coefficients of A_V and $G(s,t)$ in Table 3 will be determined by the networks chosen for A_V (see Fig. 8). The root loci for the forms of A_V are given in Fig. 9. Each of these circuits is absolutely stable.

Table 3

$$A_{V1} = H\frac{\omega_0{}^2}{s^2 + b\omega_0 s + \omega_0{}^2} \qquad G(s) = T_A\frac{-\mu(s^2+b\omega_0 s+\omega_0{}^2)}{s^2+b\omega_0 s+(1+\mu H)\omega_0{}^2}$$

$$A_{V2} = H\frac{s^2}{s^2 + b\omega_0 s + \omega_0{}^2} \qquad G(s) = T_A\frac{-\mu(s^2+b\omega_0 s+\omega_0{}^2)}{(1+\mu H)s^2+b\omega_0 s+\omega_0{}^2}$$

$$A_{V3} = H\frac{s^2 + \omega_0{}^2}{s^2 + b\omega_0 s + \omega_0{}^2} \qquad G(s) = T_A\frac{-\mu(s^2+b\omega_0 s+\omega_0{}^2)}{(1+\mu H)(s^2+\frac{b\omega_0 s}{1-\mu H}+\omega_0{}^2)}$$

Note: A_V, $G(s)$, ω_0 and μH are functions of time.

The sensitivity of the circuits using A_{V2} and A_{V1}, is similar to those of the single-loop feedback $G_1(s,t)$ and $G_3(s,t)$, forms respectively. Using the type II realization from Fig. 8, the problem of requiring a high gain from one active element can be resolved since this large gain can be distributed equally between the μ and H. Since A_{V3} and $G_2(s,t)$ have different equations, no direct comparison can be made.

An example of the multi-loop feedback S-time-varying network is shown in Fig. 10a. The A_{V1} network is of the type II form while the T_A network is chosen to give bandpass characteristics. The overall transfer function for this network is given by Eq. (6).

$$\frac{E_{out}}{E_{in}}(s,t) = \frac{-\,s\omega_0(t)}{s^2 + 3\omega_0(t)s + [1 + \mu\beta(t)][\omega_0(t)]^2} \qquad (6)$$

Fig. 10b shows the root locus of the poles for variable $\mu\beta(t)$ and constant $\omega_0(t)$. The pole-zero diagram of a 3-pole bandpass time-sharing filter is shown in Fig. 11. This filter will be used as part of a time-sharing system in Fig. 12a and has the transmission characteristics as a function of time and switch position as shown in Fig. 12b. To realize these specifications, one can choose to cascade three circuits of the form of Fig. 10a. The information required for each stage is obtained from Fig. 11 by using the following equations. (7).

$$\omega_{o1} = BW/6 \qquad \omega_{o2} = BW/3 \qquad \omega_{o3} = BW/6 \tag{7}$$

For a 3 pole-case, the $j\omega$ values are obtained from the center frequency ω_c, by Eq. (8).

$$\begin{aligned} \omega_1 &= \omega_c + 0.433\ BW \\ \omega_2 &= \omega_c \\ \omega_3 &= \omega_c - 0.433\ BW \end{aligned} \tag{8}$$

Then using the values obtained from Eqs. (7) and (8) one can find the value of the loop gain $\mu\beta_i(t)$ from Eq. (9).

$$\mu\beta_i = \frac{\left(\frac{2\omega_i}{\omega_{o\,i}}\right)^2 + 5}{4} \tag{9}$$

To find the required gain change for a certain change in frequency, one can use the relation

$$\Delta\mu\beta = \frac{\Delta(\omega^2)}{\omega_o^{\,2}} \tag{10}$$

The cascade realization for this example is shown in Fig. 13.

IV. CONCLUSION

In this paper, time-sharing active RC filters as applied to time-sharing systems have been investigated. Single-loop and multi-loop S-time varying networks have been examined. It is concluded that for lowpass (highpass) and low-Q filter realizations, single-loop S-time-varying networks can be used. For bandpass and high-Q realizations, multi-loop S-time-varying networks have been found desirable. Examples of realizations for lowpass (highpass) and bandpass time-sharing filters have been given. The practical implementation of these circuits has been investigated and the results will be given in a future paper.

REFERENCES

1. L.A. Zadeh, "Time-Varying Networks, I, "Proc. IRE, Vol. 49, Oct. 1961, pp. 1488-1503.
2. S. Darlington, "Linear Time-Varying Circuits," B.S.T.J., Nov. 1963, pp. 2575-2608.
3. C.A. Desoer and A. Paige, "Linear Time-Varying G-C Networks: Stable and Unstable", IEEE Trans. On Circuit Theory, Vol. CT - 10, June 1963, pp. 180-190.
4. R. Rohrer "Stability of Linear Time-Varying Networks - Bounds on Stored Energy," Proc. NEC, pp. 107-114, 1963.
5. I.W. Sandberg, "A Stability Criterion for Linear Networks Containing Time-Varying Capacitors," IEEE Trans. on Circuit Theory, Vol. CT 12, March 1965.
6. E.S. Kuh, "Stability of Linear Time-Varying Networks—The State Space Approach," IEEE Trans. on Circuit Theory, Vol. CT-12, June 1965, pp. 150-157.
7. J. Thorp, "Realization of Variable Active Networks," IEEE Trans. on Circuit Theory, Vol. CT-12, No. 4, Dec. 1965, pp. 511-514.
8. R.P. Sallen, and E.L. Key, "A Practical Method of Designing RC Active Filters," IRE Trans. on Circuit Theory, Vol. CT-2, March 1955, pp. 74-85.

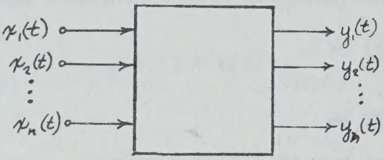

$x_1(t)$
$x_2(t)$
$x_n(t)$
$y_1(t)$
$y_2(t)$
$y_n(t)$

FIGURE 1a

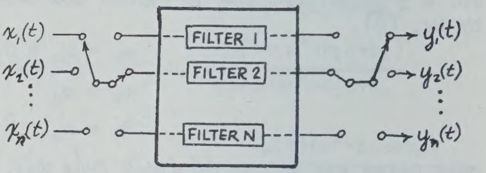

$x_1(t)$
$x_2(t)$
$x_n(t)$
FILTER 1
FILTER 2
FILTER N
$y_1(t)$
$y_2(t)$
$y_n(t)$

FIGURE 1b

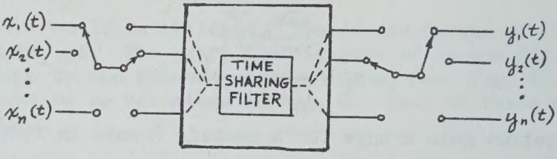

$x_1(t)$
$x_2(t)$
$x_n(t)$
TIME SHARING FILTER
$y_1(t)$
$y_2(t)$
$y_n(t)$

FIGURE 1C

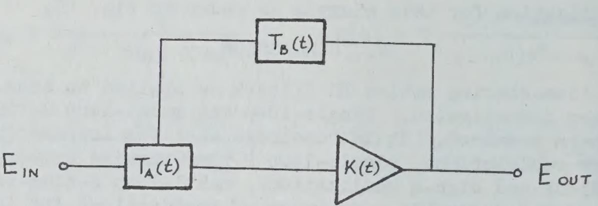
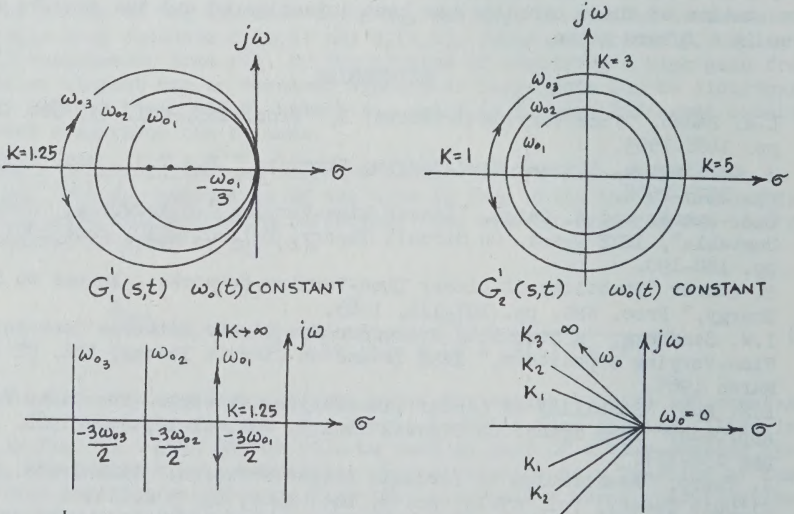

$T_B(t)$
E_{IN}
$T_A(t)$
$K(t)$
E_{OUT}

FIGURE 2


$j\omega$
ω_{o3}
ω_{o2}
ω_{o1}
K=1.25
$-\frac{\omega_{o1}}{3}$
σ
$j\omega$
K=3
ω_{o3}
ω_{o2}
ω_{o1}
K=1
K=5
σ
ω_{o3}
ω_{o2}
$K\to\infty$
ω_{o1}
$j\omega$
K=1.25
σ
$-\frac{3\omega_{o3}}{2}$
$-\frac{3\omega_{o2}}{2}$
$-\frac{3\omega_{o1}}{2}$
K_3
∞
K_2
K_1
ω_o
$j\omega$
$\omega_o = 0$
σ
K_1
K_2
K_3

$G_1'(s,t)$ $\omega_o(t)$ CONSTANT

$G_2'(s,t)$ $\omega_o(t)$ CONSTANT

$G_3'(s,t)$ $\omega_o(t)$ CONSTANT

G_1', G_2', G_3' $K(t)$ CONSTANT

FIGURE 3

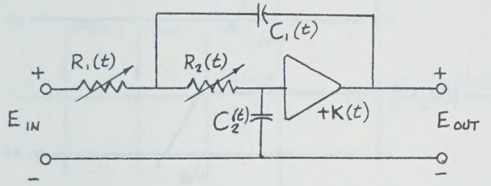

+
R₁(t)
R₂(t)
C₁(t)
E IN
C₂(t)
+K(t)
+
E OUT
-
-

FIGURE 4a

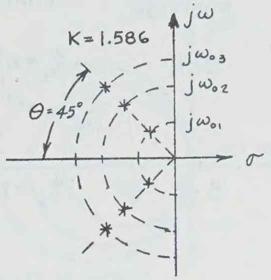

jω
K = 1.586
jω₀₃
jω₀₂
θ = 45°
jω₀₁
σ

FIGURE 4b

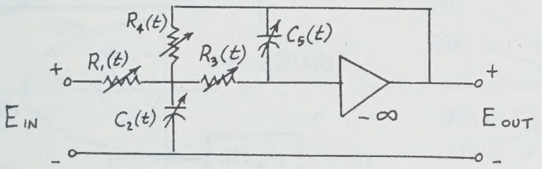

R₄(t)
C₅(t)
R₁(t)
R₃(t)
+
+
E IN
C₂(t)
-∞
E OUT
-
-

FIGURE 5a

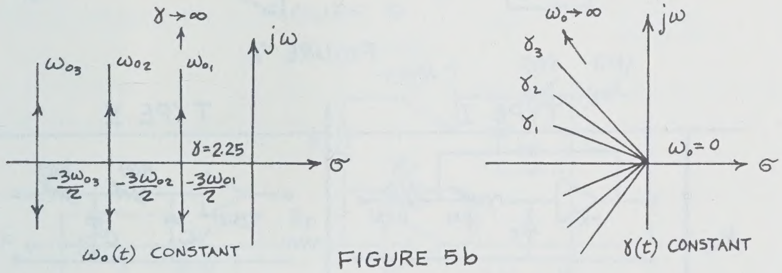

γ→∞
jω
ω₀₃
ω₀₂
ω₀₁
γ=2.25
σ
-3ω₀₃/2
-3ω₀₂/2
-3ω₀₁/2
ω₀(t) CONSTANT
ω₀→∞
jω
γ₃
γ₂
γ₁
ω₀=0
σ
γ(t) CONSTANT

FIGURE 5b

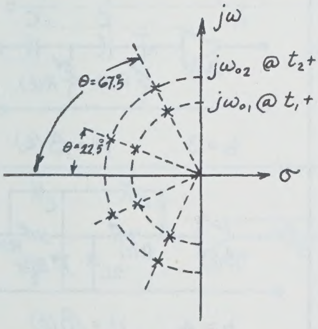

jω
θ=67.5°
jω₀₂ @ t₂+nT
jω₀₁ @ t₁+nT
θ=22.5°
σ

FIGURE 6a

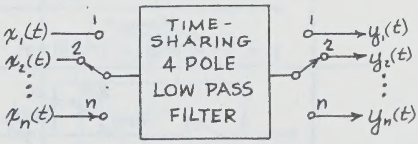

x₁(t)
x₂(t)
xₙ(t)
1
2
n
TIME-SHARING
4 POLE
LOW PASS
FILTER
1
2
n
y₁(t)
y₂(t)
yₙ(t)

FIGURE 6b

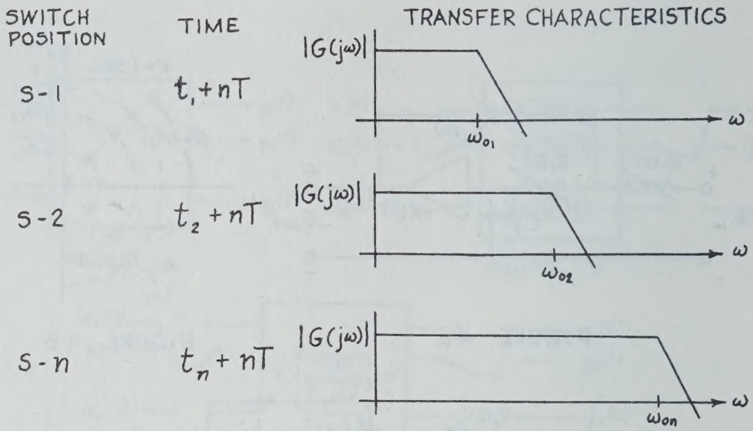


FIGURE 6C

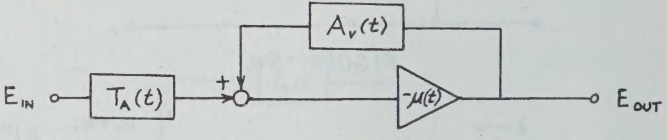

$A_v(t)$
E_{IN}
$T_A(t)$
+
$-\mu(t)$
E_{OUT}

FIGURE 7

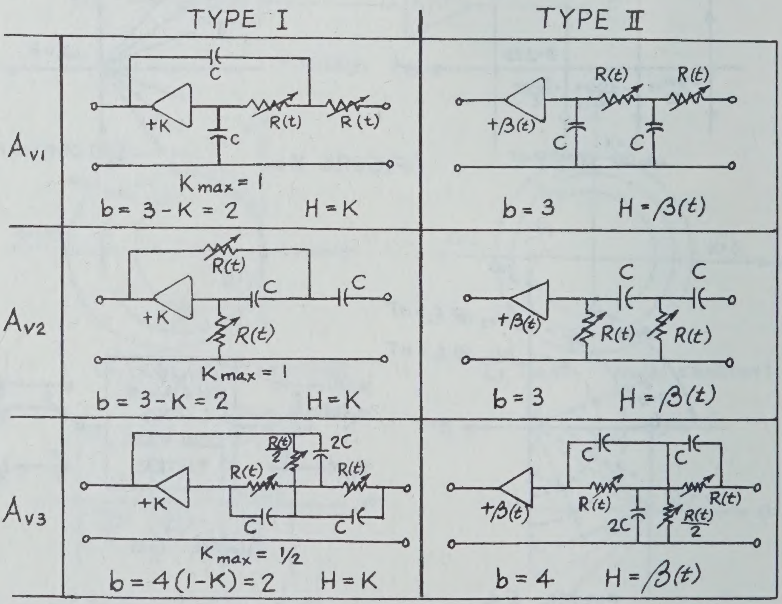
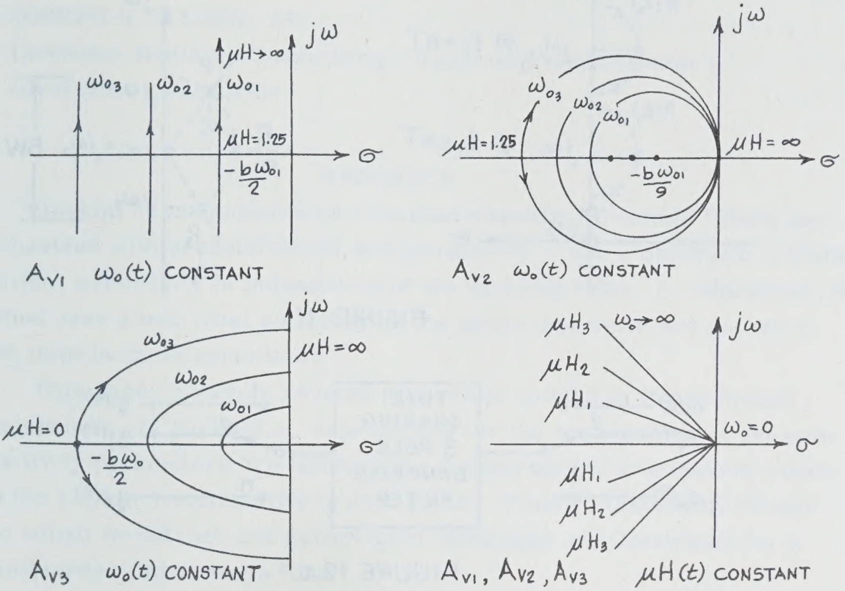

TYPE I
TYPE II
A_{V1}
C
C
+K
R(t)
R(t)
$K_{max} = 1$
$b = 3 - K = 2$
$H = K$
R(t)
R(t)
$+\beta(t)$
C
C
$b = 3$
$H = \beta(t)$
A_{V2}
R(t)
C
C
+K
R(t)
$K_{max} = 1$
$b = 3 - K = 2$
$H = K$
C
C
$+\beta(t)$
R(t)
R(t)
$b = 3$
$H = \beta(t)$
A_{V3}
$\frac{R(t)}{2}$
2C
R(t)
R(t)
+K
C
C
$K_{max} = 1/2$
$b = 4(1-K) = 2$
$H = K$
C
C
R(t)
R(t)
$+\beta(t)$
2C
$\frac{R(t)}{2}$
$b = 4$
$H = \beta(t)$

FIGURE. 8


jω
μH→∞
ω_{03}
ω_{02}
ω_{01}
μH=1.25
-bω_{01}/2
σ
jω
ω_{03}
ω_{02}
ω_{01}
μH=1.25
μH=∞
-bω_{01}/9
σ
jω
μH=∞
ω_{03}
ω_{02}
ω_{01}
μH=0
-bω_o/2
σ
μH_3
ω_o→∞
jω
μH_2
μH_1
ω_o=0
σ
μH_1
μH_2
μH_3

A_{V1} $\omega_o(t)$ CONSTANT

A_{V2} $\omega_o(t)$ CONSTANT

A_{V3} $\omega_o(t)$ CONSTANT

A_{V1}, A_{V2}, A_{V3} $\mu H(t)$ CONSTANT

FIGURE 9

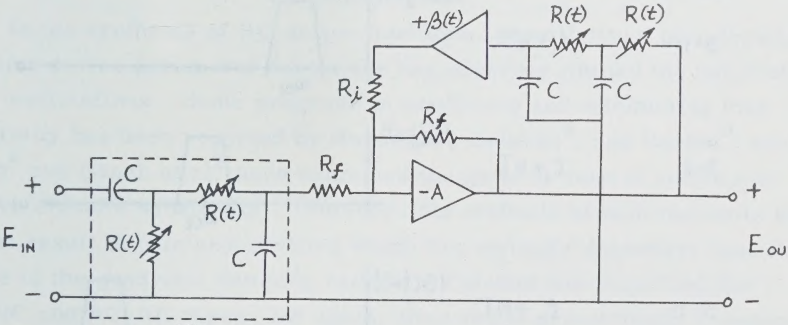

+β(t)
R(t)
R(t)
R_i
C
C
R_f
R_f
-A
C
R(t)
+
E_IN
R(t)
C
-
+
E_OUT
-
T_A

FIGURE 10a

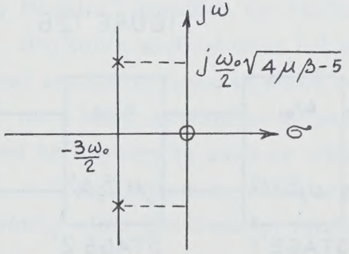

jω
$j\frac{\omega_o}{2}\sqrt{4\mu\beta-5}$
σ
$-\frac{3\omega_o}{2}$

FIGURE 10b

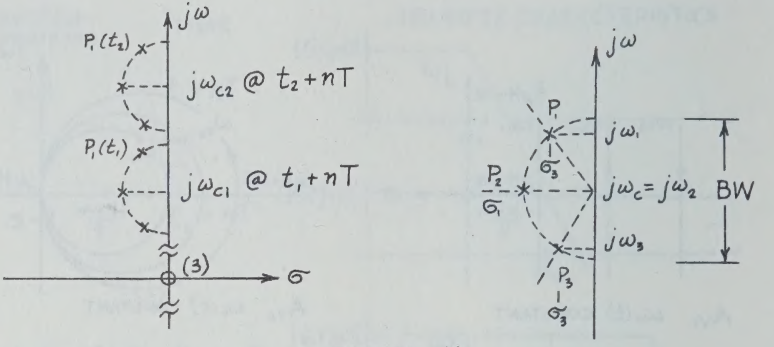

jω
$P_1(t_2)$
$j\omega_{c2}$ @ t_2+nT
$P_1(t_1)$
$j\omega_{c1}$ @ t_1+nT
(3)
σ
jω
P_1
$j\omega_1$
σ_3
P_2
σ_1
$j\omega_c = j\omega_2$
BW
$j\omega_3$
P_3
σ_3

FIGURE 11

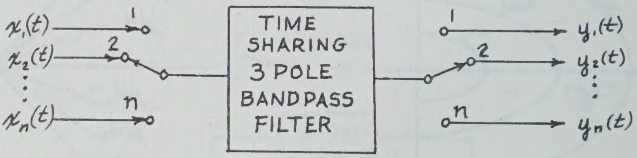

$x_1(t)$
1
$x_2(t)$
2
$x_n(t)$
n
TIME SHARING 3 POLE BANDPASS FILTER
1
$y_1(t)$
2
$y_2(t)$
n
$y_n(t)$

FIGURE 12 a

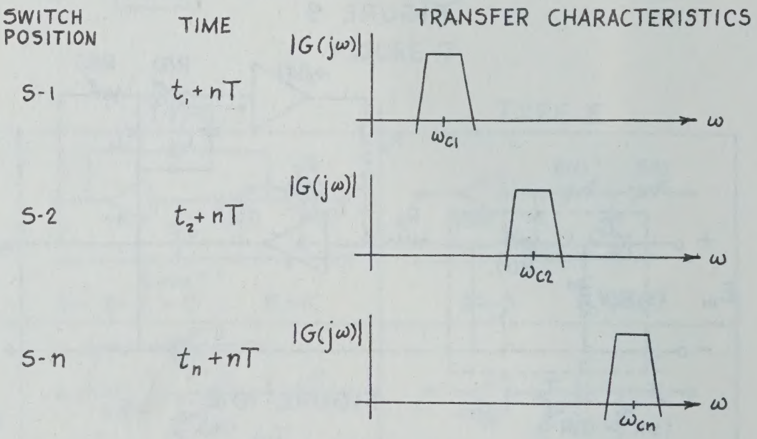


FIGURE 12b

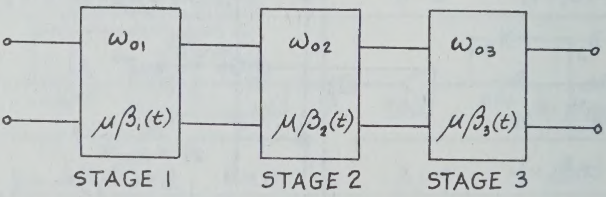

ω_{01}
$\mu\beta_1(t)$
ω_{02}
$\mu\beta_2(t)$
ω_{03}
$\mu\beta_3(t)$
STAGE 1
STAGE 2
STAGE 3

FIGURE 13

ON HAYKIM'S METHOD FOR RC-ACTIVE FILTER SYNTHESIS*

JOSEPH H. WUJEK, JR.
Lawrence Radiation Laboratory, University of California
Livermore, California

ABSTRACT

Haykim has demonstrated a method whereby RC-active filters are synthesized with predetermined pole sensitivity. For a quadratic transfer function, sensitivity is independent of the damping ratio, ζ. Moreover, the method uses a non-ideal amplifier in the sense that finite and non-zero input impedance is permitted.

This paper presents several properties related to the synthesis development. In particular, expressions for the boundedness of the pole sensitivity parameters are developed. A theorem for a necessary condition for Haykim-realizability is also given. Finally, the effects of non-zero output impedance and severe gain variations are illustrated for a second-order Butterworth filter.

INTRODUCTION

In the synthesis of RC-active functions, sensitivity of pole locations to active device parameter variations has severely limited the value of these realizations. Some progress in predicting and minimizing this sensitivity has been reported by Horowitz[1], Calahan[2], and Herbst[3]; while Gaash[4] and Gaash et al.[5] have examined the general case of active and passive element sensitivity. However, the methods of both Horowitz and Herbst result in pole sensitivities which are strongly dependent upon the choice of the quadratic damping ratio, ζ. Calahan has examined the RC-NIC and RC-RL cases, but again, the results are strongly ζ-dependent and can result in sensitivities greater than unity.

A method reported by Haykim[6] provides for realizations with the pole sensitivity predetermined. Haykim's method does not use the negative impedance converter or ideal amplifier found in some realizations. In particular, a finite and non-zero input impedance is permitted. While the output impedance is required by Haykim as zero or infinite, depending upon the nature or the amplifier, a finite, non-zero resistance amplifier output impedance does not significantly alter the transfer function character[7].

---

*Work performed under the auspices of the U.S. Atomic Energy Commission.

Properties of Haykim's Method

As a point of departure in our development, we present an inequality derived by Haykim to insure RC-realizability of a quadratic transfer function:[8]

$$\zeta^2(1 - k - mk)^2 > (1 - k)(1 - k - 2mk), \tag{1}$$

where ζ is the damping ratio, m is a measure of pole sensitivity, and k is a control parameter. We treat ζ as a parameter and rewrite (1) as a polynomial $f(m,k) > 0$, thus:

$$f(m,k) = m^2(\zeta^2 k^2) + 2m(\zeta^2 k^2 - k^2 + k - \zeta^2 k) + (k^2\zeta^2 - k^2 + 2k - 2\zeta^2 k + \zeta^2 - 1) > 0. \tag{2}$$

We shall refer to the function f(m, k) as the Realizability Function or for brevity RF, and what follows is based on the properties of the RF. Treating ζ as a parameter, for any $k \neq 0$, $0 < \zeta < 1$:

$$\frac{\partial^2 f(m,k)}{\partial m^2} = \zeta^2 k^2 > 0 \tag{3}$$

with $k \neq 0$, to be proved. Hence f(m, k) is upward concave.[9] Setting $\frac{\partial f(m,k)}{\partial m} = 0$, we find that f(m, k) has a minimum m_m at

$$m_m = (1 - 1/\zeta^2)(1/k - 1). \tag{4}$$

Now $1 - 1/\zeta^2 < 0$, since $0 < \zeta < 1$, and $1/k - 1 > 0$ since, as we shall prove, $0 < k < 1$. Hence $m_m < 0$ for all admissible values of k and ζ.

Two other properties of the RF are of interest:

1.) The roots of f(m, k), for a given k, are at:

$$m = a_i = \frac{k - 1}{\zeta^2 k}\left[1 - \zeta^2 \pm \sqrt{1 - \zeta^2}\right] \tag{5}$$

2.) The value of f(m, k) at m = 0 is:

$$f(0,k) = (\zeta^2 - 1)(k - 1)^2 < 0 \tag{6}$$

since $(k - 1)^2 > 0$, $\zeta^2 - 1 < 0$, always.

These results are illustrated in Figure 1.

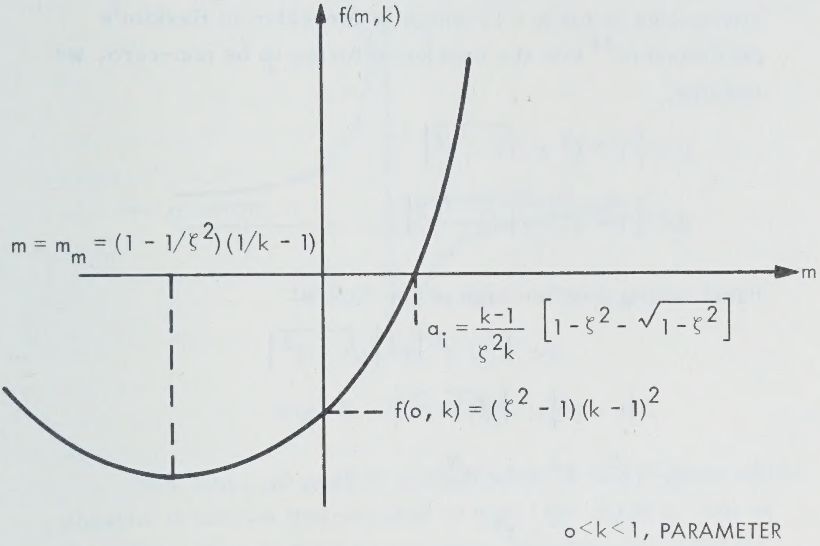


Fig. 1. The Realizability Function (RF)

Several important properties can be proved from (2) - (6).

Property (i): The parameter m is bounded by: $0 < m < 1$.

Proof If $m \geq 1$, the realization is not useful, since Haykim[10] defines:

$$\delta s_a = \delta k \left[m s_a\right] \text{ or;}$$

$$\delta s_a = \delta k \left[-m s_a^*\right] \text{ and hence the}$$

pole location moves by an increment δs_a which is identical to, or greater than, the increment δk in the control parameter. There is no loss of generality in Haykim's assumption of $m > 0$ since taking $m < 0$ results in pole movement in the opposite direction. We can account for this by taking $\delta k < 0$. We have only to show that $m \neq 0$ to complete the proof. From (5), take

$$m = 0 = \frac{k - 1}{\zeta^2 k}\left[1 - \zeta^2 \pm \sqrt{1 - \zeta^2}\right]$$

Now the denominator must be non-zero and if we can show that the terms inside the bracket are non-zero, the only alternative is for k = 1, which is forbidden in Haykim's development.[11] For the bracketed terms to be non-zero, we require,

$$0 < \left| 1 - \zeta^2 \pm \sqrt{1 - \zeta^2} \right|$$

$$0 < \left| 1 - \zeta^2 \right| \pm \left| \sqrt{1 - \zeta^2} \right| .$$

First taking positive sign of the radical:

$$0 < \left| 1 - \zeta^2 \right| + \left| \sqrt{1 - \zeta^2} \right|$$

$$- \left| 1 - \zeta^2 \right| < \left| \sqrt{1 - \zeta^2} \right|$$

$$1 - 2\zeta^2 + \zeta^4 < 1 - \zeta^2$$

$$\zeta^2 > \zeta^4$$

$$1 > \zeta^2$$

Now $\zeta < 1$, hence $\zeta^2 < 1$ follows. Since the radical is squared in the above, the result holds for the negative root as well, Q.E.D..

Property (ii): The control parameter k is bounded by $0 < k < 1$.

Proof As we shall see, it is advantageous to restrict the lower bound on k in this manner.

For k = 0, the condition (2) is violated, since k = 0 requires $\zeta^2 - 1 > 0$, which is inadmissable. Now even in the weakest case, Haykim requires $k < 1$, so we need investigate only the possibility of $k < 0$.

Let us rewrite the solution for the roots of the RF in (5) as:

$$a_i = \left[\frac{k - 1}{k} \right] \left[\frac{1 - \zeta^2}{\zeta^2} \right] \left[\frac{\pm \sqrt{1 - \zeta^2}}{\zeta^2} \right] .$$

Now only the positive root is of interest since $0 < m < 1$ is the range of m. For insight into the behavior of the roots of the RF, we plot $\frac{k - 1}{k}$ in Figure 2.

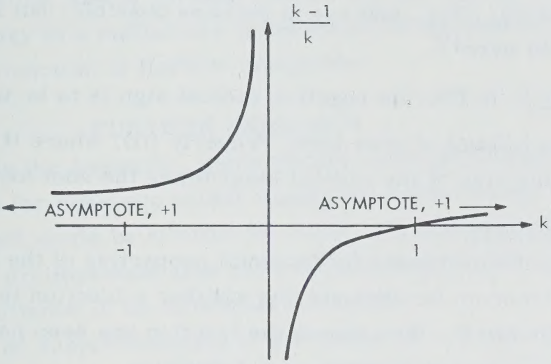


Fig. 2. Plot of $\frac{k-1}{k}$

Now since we seek to minimize m for any ζ given and k chosen, it follows that we seek to make the positive root of the RF as close to zero as possible, since the RF must be positive by (2) if a realization is to result. Examination of (6) shows that the appropriate sign of the radical is taken to secure the positive root, i.e.:

(a) Minus sign is taken if $0 < k < 1$.

(b) Plus sign is taken if $k < 0$.

The result (a) and (b) follow from Figure 2 and

$$0 < \frac{1-\zeta^2}{\zeta^2} < 1 \ .$$

If now we take $f(k) = \frac{k-1}{k}$; and since we are interested only in magnitudes, take

$$\left|\frac{1-\zeta^2}{\zeta^2}\right| \cdot \left|\frac{1-\zeta^2}{\zeta^2}\right| = M \ ,$$

then (7) may be written:

$$a_i = |f(k)|\, M \ .$$

Clearly, if a_i is to be made minimum, from Figure 2 we choose $0 < k < 1$. This same result may be obtained analytically by use of inequalities and $|f(k)|$. A proof by means of application of the Fialkow-Gerst conditions to Haykim's formulas for y_{21}, y_{11}, and y_{12} is perhaps possible, but is not attempted here[12].

Property (iii): In (5), the negative radical sign is to be taken.

Proof This follows at once from Property (ii), where it was shown that the sign of the radical determines the root location.

A Realizability Theorem

Having thus obtained some fundamental properties of the RF, m, and k, we now cite a theorem for determining whether a function is Haykim-realizable for a chosen ζ. We assume the function has been normalized to ζ, and refer to the resultant function as a ζ-function.

Theorem

A necessary condition for a ζ-function to be Haykim-realizable is that for the given ζ and chosen m_i and k_i, with $0 < m_i < 1$ and k_i chosen, is that[13]

$a_i < m_i$, where

$$a_i = \frac{k_i - 1}{k_i \zeta^2} \left[1 - \zeta^2 - \sqrt{1 - \zeta^2} \right]. \tag{9}$$

Proof

If $m_i \leq a_i$, then (2) is violated, and hence y_{11} will not be RC-realizable. That (2) will indeed be violated by $m_i \leq a_i$ is verified by examination of the RF in Figure 1, or by direct substitution of k_i and m_i in (2). Since the RF is quadratic in m, only one finite maximum or minimum occurs, and this is a minimum as given in (4). Moreover, this minimum occurs at a negative value of m, and the RF must also be negative there, since the RF is upward concave by (3) and possesses real roots.

This theorem may be thought of as the Strong Bounds Law of m_i and k_i, whereas Properties (i) and (ii) furnish the Weak Bounds Law. The theorem also facilitates trade-off studies of "best" choice of k_i vs m_i. The physical significance of k_i remains somewhat obscure, except to note that the permissible region for k becomes vanishingly small as $m_i \rightarrow a_i$ from the right. Further study in this area is required.

The use of (9) greatly simplifies choice of k_i. The significance of naming (2) the Realizability Function is now evident. The location of the positive root of the RF provides a lower bound on the m_i which may be realized. The theorem cited appears to contain all the requirements for

Haykim-realizability, and is perhaps a sufficient as well as necessary condition. A corollary to a sufficiency proof would be the question of whether any realizable ξ-function is Haykim-realizable.

FURTHER RESEARCH

As shown in the Appendix, there results some perturbation in the transfer function for non-zero output resistance in the voltage amplifier. Clearly, the effect would be similar for finite output admittance of a current amplifier. An important area of research would center on the inclusion of this immittance in the synthesis procedure.

Extension or supplementation of the method to include synthesis in the general case of more than two complex poles and more than one real pole without the use of isolation amplifiers would also be useful. Ideally included in such a procedure would be a means whereby poles are weighted in terms of the order of their contribution in obtaining the specified frequency response.

APPENDIX

POLE-MOVEMENT FOR SEVERE VARIATION IN μ WITH NON-ZERO OUTPUT RESISTANCE

Haykim's paper gives two examples using a voltage amplifier having zero output impedance, and the pole-movement is examined for a ±10% change in μ. In this section we shall add a non-zero resistance in the amplifier output, and observe the change in transfer function character for ±100% variations in μ.

Haykim gives a realization of the second-order Butterworth filter:

$$T_{12} = \frac{I_2}{I_1} = \frac{K(0)}{s^2 + \sqrt{2}\, s + 1} .$$

The realization is shown in Figure A1. The 3.69 ohm resistor across the amplifier input is assumed to be the combined resistance of the bias network and amplifier input resistance. If we require an amplifier to have an input to output impedance transformation of 100 to 1, which is easily obtained, the voltage amplifier is modified by taking R = 0.0369. The voltage-controlled current source for purposes of computation.[14] The presence of the non-zero amplifier output resistance nullifies the pole-zero cancellation which occurs for zero output impedance. Figure A2 compares the pole-zero plots of T_{12} for Haykim's realization when μ varies ±100% from the design value, and when we take $R \neq 0$ as well as $R = 0$. The near-cancellation of the poles and zeros on the negative real-axis does not materially alter the frequency response from the perfect cancellation case, as expected[15]. Hence, the presence of a non-zero output resistance is allowed in this case, and a practical amplifier may be used in the realization. The use of Darlington's compound transistor in common-collector configuration may be used as an output stage, while conventional common-emitter input stage(s) furnish the desired voltage gain and 180° phase-shift. Thus, impedance transformations of 10,000 are obtained. Use of a field effect transistor in the input stage yields transformations of 10^{10} and greater.

ACKNOWLEDGMENT

The author is grateful to W. G. Magnuson of the Lawrence Radiation Laboratory for providing a source-deck of Calahan's computer program and valuable instruction in its use. The encouragement provided by Professor A. S. McAllister of San Jose State College and Professor D. O. Pederson of the University of California is also appreciated.

REFERENCES

[1] Horowitz, I. M., Optimization of Negative Impedance Methods of Active RC Synthesis, IRE Trans., CT-6, p. 296.

[2] Calahan, D. A., Sensitivity Minimization in Active RC Synthesis, IRE Trans., CT-9, p. 38.

[3] Herbst, N. M., Optimization of Pole Sensitivity in Active RC Networks, Cornell University, Research Report EE 569, 1963.

[4] Gaash, A. A., Synthesis of Integrated Selective Amplifiers for Specified Response and Desensitivity, Report 65-31, Electronics Research Lab., University of California (Berkeley), June 1965.

[5] Gaash, A. A., Pepper, R. S., and Pederson, D. O., Design of Integrable Desensitized Frequency Selective Amplifiers, IEEE Journal of Solid State Circuits, SC-1, No. 1, Sept. 1966, pp. 29-35.

[6] Haykim, S. S., Synthesis of RC Active Filters with Prescribed Pole Sensitivity, Proc. IEE (London), Vol. 112, No. 12, Dec. 1965, p. 2235.

[7] Cf. Appendix.

[8] Haykim, op. cit.

[9] Cf. any calculus text, e.g. Thomas, G. B., Calculus and Analytic Geometry, Addison-Wesley Co., Reading, Mass., 1956, pp. 111-13.

[10] Haykim, op. cit.

[11] Haykim, ibid.

[12] Fialkow, A. D. and Gerst, I., The Transfer Function of General Two Terminal - Pair RC Networks, Quart. Appl. Math, Vol. 10, 1953, p. 113.

[13] An error appears in Haykim's paper regarding permissible values of k. Cf. correction Proc. IEE (London), Correspondence, V. 114, No. 1, Jan. 1967, pp. 85-86.

[14] Calahan, D. A., Linear Network Analysis and Realization Digital Computer Programs, An Instruction Manual, University of Illinois, Engineering Experiment Station Bulletin 472, Feb. 1965.

[15] Pederson, D. O. and Wilson, G. H.; Dominant Zero and Excess Phase of a Hurwitz Polynomial, IEEE Trans. Circuit Theory, CT-11, March 1964, pp. 104-108.

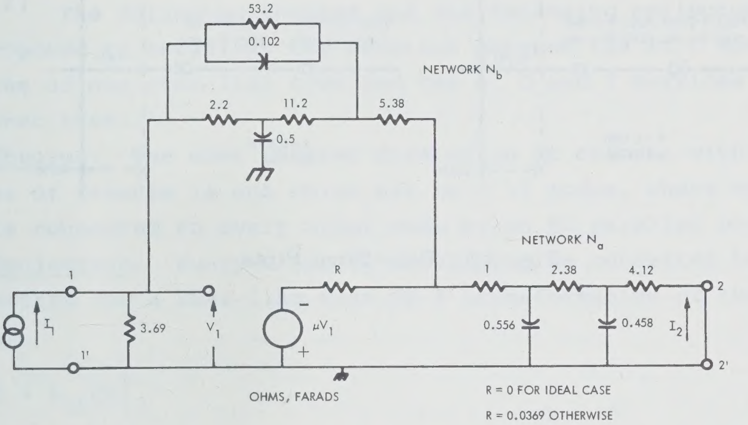

53.2
0.102
NETWORK N_b
2.2
11.2
5.38
0.5
NETWORK N_a
1
R
1
2.38
4.12
2
I_1
3.69
V_1
μV_1
0.556
0.458
I_2
1'
2'
OHMS, FARADS
R = 0 FOR IDEAL CASE
R = 0.0369 OTHERWISE

Fig. A1. Haykim Realization

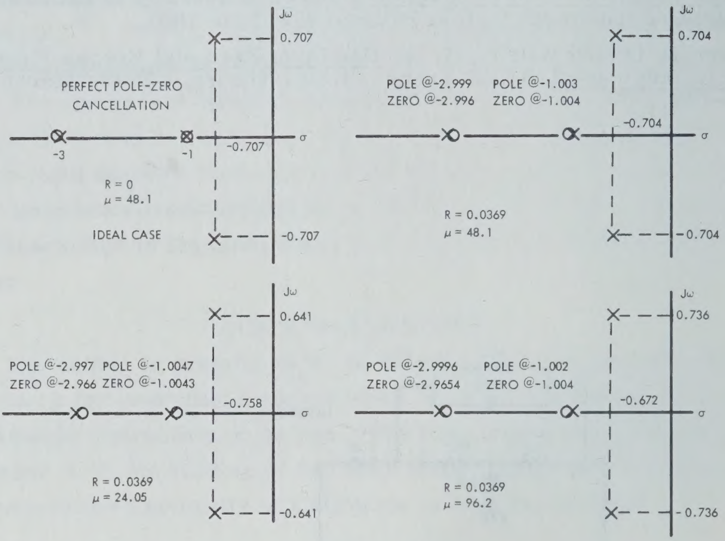

Jω
0.707
PERFECT POLE-ZERO
CANCELLATION
-3
-1
-0.707
σ
R = 0
μ = 48.1
IDEAL CASE
-0.707
Jω
0.704
POLE @-2.999
ZERO @-2.996
POLE @-1.003
ZERO @-1.004
-0.704
σ
R = 0.0369
μ = 48.1
-0.704
Jω
0.641
POLE @-2.997
ZERO @-2.966
POLE @-1.0047
ZERO @-1.0043
-0.758
σ
R = 0.0369
μ = 24.05
-0.641
Jω
0.736
POLE @-2.9996
ZERO @-2.9654
POLE @-1.002
ZERO @-1.004
-0.672
σ
R = 0.0369
μ = 96.2
- 0.736

Fig. A2. Pole-Zero Plots

SYNTHESIS OF BASHKOW'S A MATRIX FOR THE GENERAL RC CASE*

J. E. STEELMAN AND S. KARNI

Bureau of Engineering Research
The University of New Mexico
Albuquerque, New Mexico

Abstract

For many uses, a convenient formulation for solving a circuit is Bashkow's A matrix.$^{(1)}$ To find Bashkow's A matrix for a given RC circuit, a tree must be chosen which contains the maximum number of capacitances. Such a tree is called a proper tree.$^{(2)}$ The following theorem and the following conjecture are proposed to establish the relation between the A, C and Y matrices of one star-like tree and the A, C and Y matrices of any other tree.

Theorem: The most general form of an RC circuit with n degrees of freedom is one which has (n + 1) nodes, where each node is connected to every other node by an RC parallel pair.

Conjecture: Every C (or Y) matrix can be converted to the C(Y) matrix for a star-like tree by a transformation of the form

$$\hat{C} = A_{12} C A_{12}^T$$

where A, C and Y are related by $A = -C^{-1}Y$

C is the cut-set capacitance matrix for those fundamental cut-sets defined by capacitive tree branches;

Y is the short-circuit admittance matrix of the purely resistive multiport network formed by the tree branch capacitances when all of the remaining capacitive links are open circuited;

$\hat{C}$ is the C matrix for a star-like tree;

A_{12} is a nonsingular submatrix of the incidence matrix.

Now, C and Y are both positive definite matrices; therefore, it is possible to find matrix P_1 such that

$$P_1^T C P_1 = I \text{ (I = the unit matrix)}$$

---

*This research is supported by NSF Grant No. GK-1380.

and

$$P_1^T Y P_1 = \Lambda_1$$

The eigenvalues Λ_1 of C and Y are related to the eigenvalues Λ_2 of A by $\Lambda_1 = -\Lambda_2$. Further, the eigenvectors of C and Y and the eigenvectors of A are the same within a constant length multiplier. This leads to the following two equations for $\hat{Y}$ and $\hat{C}$.

$$\hat{Y} = A_{12}(P^T)^{-1} D^{-2}(-\Lambda_2)P^{-1}A_{12}^T$$

$$\hat{C} = A_{12}(P^T)^{-1} D^{-2}P^{-1}A_{12}^T$$

where

P is a diagonalizing matrix for A and is known;

A_{12} is a submatrix of the incidence matrix and is unknown;

D^{-2} is a diagonal matrix and is unknown;

Λ_2 is the matrix of eigenvalues of A and is known.

Dominance is a necessary and sufficient condition for realizing $\hat{C}$ and $\hat{Y}$.(3) Expansion of the expressions for C and Y leads to all possible terms of the form $a_{ij}a_{kl}d_i^{-2}m$, where a_{ij} and a_{kl} are unknown elements of A_{12} and d_i^{-2} is an unknown element of D^{-2}. By treating terms of the form $a_{ij}a_{kl}d_i^{-2}$ as new unknowns, the dominance requirement may be put into a format of linear programming.(4) Such an approach requires that a number of other inequalities be satisfied. For a (2 x 2)A matrix this procedure leads to 38 equalities and inequalities in 26 unknowns. A computer solution (on a CDC 3600) required 22 seconds.

A geometrical interpretation of linear programming is useful for this problem. The equalities and inequalities specify a convex region in a hyperspace. If a solution exists, linear programming finds a set of positive numbers, i.e., a point in the hyperspace; then finds the point which minimizes a given objective function. The linear programming algorithm moves the solution point from corner to corner of the convex region while constantly decreasing the objective function.(4)

The (2 x 2) case has been successfully synthesized using this procedure. Future work will apply the procedure to higher order cases. Future work will also determine the best general objective function (or sequence of functions).

References

1 Bashkow, T. R.: "The A matrix, new network description", IRE Trans on Circuit Theory, 1957, CT-4, p. 117.

2 Kuh, E. S., and Rohrer, R. A.: "The state variable approach to network analysis", Proc. Inst. Elect. Electronics Engrs, 1965, 53, p. 672.

3 Slepian, P., and Weinberg, L.: "Synthesis applications of paramount and dominant matrices", Pro NEC, 1958, 14.

4 Dantzig, G. B.: "Linear programming and extensions" (Princeton, 1963).

AN OPTIMAL FIXED CONTROL STRUCTURE DESIGN WITH MINIMAL SENSITIVITY FOR A LARGE ELASTIC BOOSTER

THOMAS HENDRICKS
Martin Marietta Corp., Aerospace Div.
Denver, Colorado

HENRY D'ANGELO
Denver University
Denver, Colorado

ABSTRACT

A design algorithm developed for the minimization of a cost functional which includes, in addition to state and control variables, a measure of sensitivity, is used to solve an example of practical interest: a flexible ballistic missile in powered flight. Comparisons are made between optimal designs which include sensitivity in the performance index and those that do not. The efficacy of using a measure of sensitivity in the performance index is demonstrated.

I. INTRODUCTION

Trajectory sensitivity, defined as $v = \partial x/\partial \alpha$ where x is the system state, α a system parameter, was introduced and used by Miller and Murray [1] in a mathematical error analysis of differential analyzers. Meissinger [2] used trajectory sensitivity functions in control system analysis in 1960. And in 1964 Kokotovic [3] demonstrated how sensitivity functions can be simulated in a straight forward manner on an analog computer. An early use of trajectory sensitivity functions in an optimal design was formulated, but not solved, by Siljak and Dorf [4]. Methods for the minimization of a performance index containing sensitivity terms have been advanced by numerous authors [5, 6, 7]. In the minimization of a performance index containing sensitivity functions the optimal control law, if feedback control is used, will in general be a function of these sensitivity variables. In this case, it is known [8] that the sensitivity functions can only be approximately realized. To avoid approximating sensitivity functions, the approach taken here is to optimize under the constraint that the control be a linear function of only the states, i.e., a constant feedback structure is assumed. The constant feedback structure is easiest to both implement and handle mathematically. Obviously, other feedback structures could also be assumed. However, for each case it would be necessary to determine if an optimal solution is independent of initial conditions. The minimization of a performance index containing sensitivity with the added constraint of fixed feedback structure is illustrated by solving a significant problem of practical interest: a fifth order flexible booster in powered flight. It is shown that if the trajectory dispersions, as system parameters are varied, are used to evaluate two optimal designs (one where sensitivity is used and one where sensitivity is not) that the design with sensitivity performs in a superior manner.

II. STATEMENT OF THE PROBLEM

The plant to be controlled is a large Titan III flexible booster. It is assumed that the states of the system are exactly known, i.e. the sensors are noiseless and contain no dynamics. Further, it is desired to compare designs by the optimum regulator theory with designs in which the performance index contains sensitivity terms. The booster is described by a set of nonlinear differential equations linearized about some nominal value. Control is maintained by gimballing the two engines which produce the thrust for the vehicle. Gimballing dynamics are neglected. Rigid and lateral elastic bending geometry is shown in Figure 1 and 2 respectively. Two frames of reference are used in describing the motion of the center of gravity of the vehicle: the inertial reference fixed on earth and the vehicle body coordinates. The relationship between the two coordinate systems is given in Figure 1. The nonlinear equations describing the motion of the center of gravity (CG) of the vehicle are written in terms of θ, α, and η (attitude, angle of attack, and bending mode deflection).

$$\ddot{\theta} I = -TL_G \sin\delta + L_n q S_f C_{m_\alpha} \alpha + \varphi_{aft} \ddot{\eta}$$

$$Mv_o (\dot{\theta} - \dot{\alpha}) = T \sin\delta + qS_f C_{n_\alpha} \alpha \qquad (1)$$

$$\ddot{\eta} = -2\zeta\omega\dot{\eta} - \omega^2\eta + Th_g \delta$$

By defining coefficients a_i by

$$a_1 = T/M \qquad a_2 = (qS_f C_{n_\alpha}/M) \qquad a_3 = v_o$$

$$a_4 = (qS_f C_{m_\alpha} L_n/I) \qquad a_5 = (TL_G/I)$$

with T = Thrust, M = Mass, $q = 1/2\,\rho v_o^2$, S_f = Reference vehicle area, L_n = moment arm, C_{n_α} = normal force coefficient C_{m_α} = aerodynamic moment coefficient, L_G = normal coordinate to normal force moment arm, g = gravity, v_o = velocity, φ_{aft} = maximum angular deflection from rigid body center line, δ = engine angular deflection, ω = bending mode natural frequency, ζ = bending mode damping ratio, h_g = bending moment arm, ρ = air density; and using the small angle approximation $\delta \cong \sin\delta$, enables Eqn. (1) to be rewritten in state variable form.

$$\frac{d}{dt}\begin{bmatrix} \theta \\ \dot{\theta} \\ \alpha \\ \eta \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & a_4 & -\omega^2\varphi_{aft} & -2\zeta\omega\varphi_{aft} \\ 0 & 1 & -a_2/a_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -\omega^2 & -2\zeta\omega \end{bmatrix}\begin{bmatrix} \theta \\ \dot{\theta} \\ \alpha \\ \eta \\ \dot{\eta} \end{bmatrix} + \begin{bmatrix} 0 \\ a_5 + \varphi_{aft}\, Th_g \\ -a_1/a_3 \\ 0 \\ Th_g \end{bmatrix}\delta \qquad (2)$$

Choosing δ (the engine deflection) to minimize the performance index

$$\Phi = \frac{1}{2}\int_0^\infty (x\cdot Qx + \delta^2)dt \qquad (3)$$

results in a linear constant feedback law $\delta = +\, g^*x$ (*denotes matrix transposition and $x\cdot Qx \equiv x^*Qx$) where g is obtained as a function of the solution to the nonlinear matrix Riccati equation

$$A^*K + KA - KBR^{-1}B^*K + Q = 0 \tag{4}$$

namely

$$g^* = -R^{-1}B^*K \tag{5}$$

The selection of δ to minimize a performance index

$$\Phi = \frac{1}{2}\int_0^\infty (x \cdot Qx + v \cdot Sv + \delta^2)dt \tag{6}$$

containing, in addition to "state" and "control", a measure of trajectory sensitivity is effected by an iterative algebraic algorithm developed in [9] under the additional constraint that the feedback law is a linear function of the state variables. The algorithm presented in [9] is a generalization, and a computational improvement, of the work of Schoenberger [10].

For notational simplicity the system to be controlled is written in the following form

$$\mathring{x} = Ax + b\delta \tag{7}$$

The control is assumed to be a constant function of the state variables

$$\delta = g^*x \tag{8}$$

Combining equations (7) and (8) yields the compensated system

$$\mathring{x} = (A + bg^*)x \overset{\Delta}{=} \tilde{A}x \tag{9}$$

and the sensitivity equations, with respect to α, with equation (9) is

$$\mathring{v} = \tilde{A}_\alpha x + \tilde{A}v \qquad v(0) = 0 \quad , \tilde{A}_\alpha \overset{\Delta}{=} \partial\tilde{A}/\partial\alpha \tag{10}$$

In block form equations (9) and (10) become

$$\begin{bmatrix} \mathring{x} \\ \mathring{v} \end{bmatrix} = \begin{bmatrix} \tilde{A} & 0 \\ \tilde{A}_\alpha & \tilde{A} \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} \tag{11}$$

To simplify equation (11) we define

$$y = \begin{bmatrix} x \\ v \end{bmatrix} \qquad \bar{A} = \begin{bmatrix} \tilde{A} & 0 \\ \tilde{A}_\alpha & \tilde{A} \end{bmatrix} \qquad W = \begin{bmatrix} Q + gg^* & 0 \\ 0 & S \end{bmatrix}$$

Then using the above nomenclature equation (11) and the performance index equation (6) can be rewritten

$$\mathring{y} = \bar{A}y \tag{12}$$

$$\Phi = \frac{1}{2}\int_0^\infty (y \cdot Wy)dt \tag{13}$$

The minimization of equation (6) requires iteratively computing the gain vector g by the formula

$$g^{\nu+1} = \{P^{\nu}\}^{-1} \{d^{\nu} + e^{\nu}\} \tag{14}$$

which is terminated when

$$\left\| g^{\nu+1} - g^{\nu} \right\| \leq \epsilon \tag{15}$$

where ϵ is a small positive number. To solve for P, d, and g of Eq. (7) requires, at each iteration, solving two 2nth order algebraic equations.

$$\bar{A}^*H + H\bar{A} + W = 0 \tag{16}$$

and

$$\hat{H}\bar{A}^* + \bar{A}\hat{H} + \begin{bmatrix} x_o x_o^* & 0 \\ 0 & 0 \end{bmatrix} = 0 \tag{17}$$

Matrices H and $\hat{H}$ are symmetric and positive definite. From the partitions of H and $\hat{H}$, i.e.

$$H = \begin{bmatrix} H_1 & H_2 \\ H_2^* & H_3 \end{bmatrix} \tag{18}$$

$$\hat{H} = \begin{bmatrix} \hat{H}_1 & \hat{H}_2 \\ \hat{H}_2^* & \hat{H}_3 \end{bmatrix} \tag{19}$$

it is possible to solve for the matrices of the algorithm from the relationships

$$P = \hat{H}_1$$

$$d = (\hat{H}_1 H_1 + \hat{H}_2 H_2^*)b \tag{20}$$

$$e = (\hat{H}_2^* H_2 + \hat{H}_3 H_3)b$$

Since Eqn. (10) definitely depends upon the initial conditions it is natural to assume that the control law $\delta = g^*x$ likewise is a function of the initial conditions. It can be shown that this is not the case; numerous numerical examples with varying initial conditions demonstrate the nondependence of the control law upon initial conditions.

III. SOLUTION OF THE PROBLEM

As a numerical example consider the parameters for a Titan III - MOL configuration at 60 seconds after liftoff. The model given in equation (2) is used; the a_i are

$a_1 = 904.8 \qquad a_2 = 213.5 \qquad a_3 = 218$

$a_4 = .3358 \qquad a_5 = -2.942$

Using the numerical values given above in equation (2) yields

$$\frac{d}{dt}\begin{bmatrix}\theta\\ \dot{\theta}\\ \alpha\\ \eta\\ \dot{\eta}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0 & 0\\ 0 & 0 & -1.5\text{x}10^{-5} & -17.5\text{x}10^{-4} & -4.4\text{x}10^{-6}\\ 0 & 1 & -9.76\text{x}10^{-3} & 0 & 0\\ 0 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & -40.2 & -0.102\end{bmatrix}\begin{bmatrix}\theta\\ \dot{\theta}\\ \alpha\\ \eta\\ \dot{\eta}\end{bmatrix} + \begin{bmatrix}0\\ -1.67\\ -0.0414\\ 0\\ -2.9\text{x}10^{4}\end{bmatrix}\delta \quad (21)$$

Of all the parameters defined, experience has indicated that the parameter ϕ_{aft} (shown in equation (2)) is not accurately known and also greatly affects system performance. Using the algorithm shown in Section II the system performance is compared for an optimum system with and without sensitivity added to the performance index.

For the optimum regulator problem the control variable is chosen to minimize the performance index

$$\Phi = \frac{1}{2}\int_0^{\infty}(x^* Qx + \delta^2)dt \qquad (22)$$

where x is defined as $x = [\theta \,,\, \dot{\theta} \,,\, \alpha \,,\, \eta \,,\, \dot{\eta}\,]^*$
and

$$Q = \begin{bmatrix}124 & 9 & 0 & 0 & 0\\ 9 & 3 & 0 & 0 & 0\\ 0 & 0 & 3 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0\end{bmatrix} \qquad (23)$$

The optimal control is found to be

$$\delta = -11.1\theta - 7.2\dot{\theta} - 0.848\alpha + 2.49 \times 10^{-4}\eta + 2.17 \times 10^{-4}\dot{\eta} \quad (24)$$

To include a measure of sensitivity in the performance index define the sensitivity variable as

$$v = \frac{\partial x}{\partial \varphi_{aft}} \qquad (25)$$

Thus the optimal control problem then is to pick the control function δ, constrained to be a constant linear function of the state variables, to minimize the performance index

$$\Phi = \frac{1}{2}\int_0^\infty (x\cdot Qx + v\cdot Sv + \delta^2)dt \qquad (26)$$

where Q has the same numerical value as before and

$$S = \text{Diag.}\left[1x10^{-16},\ 1x10^{-16},\ 1x10^{-16},\ 1x10^{-16},\ 1x10^{-16}\right]$$

The algorithm of Section II was programmed on a CDC6400 computer and the control function was found to be

$$\delta = -13.07\,\theta - 10.04\,\dot{\theta} - 1.95\,\alpha - 1.4x10^{-3}\,\eta - 2.4x10^{-5}\,\dot{\eta} \qquad (27)$$

Figure 3 shows the state space trajectories of the optimum system without sensitivity, whereas Figure 4 depicts the state space trajectories of the optimum system with sensitivity. Each figure displays three trajectories obtained by varying φ_{aft} from its nominal value + 20%. The input into the system was a step function of negative unit height on the system attitude (θ). During an actual flight this might correspond to a wind gust on wind shear input. The magnitude of the input, however, was used only for purposes illustration. All other state variables are assumed to have zero initial conditions. It is obvious from Figure 3 that the system for $\varphi_{aft} = \varphi_{aft(low)}$ is unstable and the system's trajectories diverge with increasing time. Figure 4 depicts trajectories of the optimum system with sensitivity for the same variations in φ_{aft}. In all cases the system remains stable. Also, the dispersion of the system's trajectories is not great and the utility of minimizing a functional containing sensitivity is demonstrated. It should be mentioned that this method of comparing system performance for parameter variations is not without flaws. To obtain the insensitive performance requires larger feedback gains and more control effort. It is theoretically possible, in the optimum regulator problem, to choose the weighting matrix Q so that the feedback gains will match those shown in equation (27). However, the utility of including sensitivity in the performance is one of having analytical control over the degree of insensitivity required, and achieving insensitivity in an optimum manner.

REFERENCES

1. Miller, K. S. and F. J. Murray, "A Mathematical Basis for an Error Analysis of Differential Analyzers", J. Math. and Phys., No. 3, Vol. 32, 1953, pp. 136-163.

2. Meissinger, H. F., "The Use of Parameter Influence Coefficients in Computer Analysis of Dynamic Systems", Proc. of the Western Jt. Computer Conference, 1960, pp. 181-192.

3. Kokotovic, P., "The Sensitivity Point Method in the Investigation and Optimization of Linear Control Systems", Avtomatika i Telemekhanika, Vol. 25, No. 12, 1964.

4. Siljak, D. D., and R. Dorf, "On the Minimization of Sensitivity in Optimal Control Systems", Proceedings of the Third Annual Allerton Conference, 1965, pp. 225-229.

5. Tuel, W., "Optimal Control of Unknown Systems", Ph.D. Thesis, Rensselaer Poly. Inst., June 1965.

6. Gavrilovic, M., and R. Petrovic, "On the Synthesis of the Least Sensitive Control", Proc. of International Symposium, Dubrovnik, 1964.

7. D Angelo, H. M. Moe, and T. Hendricks, "Trajectory Sensitivity of an Optimal Control System", Proceedings 4th Annual Allerton Conference, October 1966.

8. Kreindler, E., "Synthesis of Flight Control Systems Subject to Vehicle Parameter Variations", Technical Report AFFDL-TR-66-209, December 1966.

9. Hendricks, T., "Trajectory Sensitivity of an Optimal Control System of Fixed Structure", Ph.D. Thesis, Denver University, May 1967.

10. Schoenberger, M., "System Optimization with Fixed Control Structure", Ph.D. Thesis, Illinois University, January 1966.

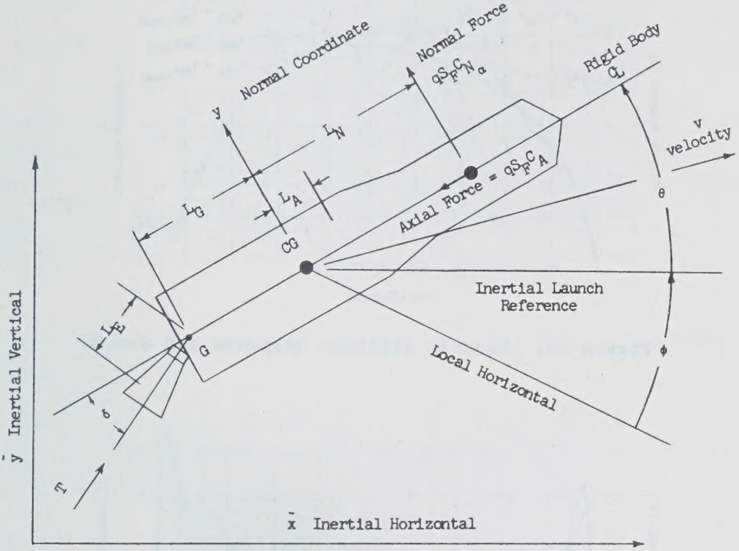
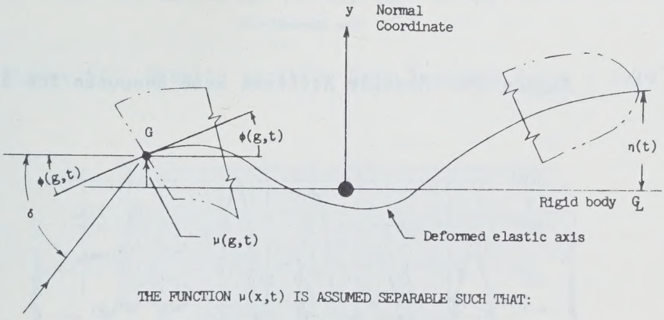

Normal Coordinate
Normal Force
$qS_FC_{N_\alpha}$
Rigid Body
℄
y
L_N
v
velocity
L_G
L_A
Axial Force = qS_FC_A
θ
CG
Inertial Launch
Reference
$\bar{y}$ Inertial Vertical
L_E
G
Local Horizontal
ϕ
δ
T
$\bar{x}$ Inertial Horizontal

Figure 1. Rigid Body Missile Geometry


y
Normal
Coordinate
G
ϕ(g,t)
ϕ(g,t)
δ
η(t)
Rigid body ℄
μ(g,t)
Deformed elastic axis
T

THE FUNCTION μ(x,t) IS ASSUMED SEPARABLE SUCH THAT:

$$\mu(x,t) = h_g(x)\eta(t)$$

WHERE $h_g(x)$ IS SET EQUAL TO UNITY AT THE NOSE.

THE SLOPE OF THE DEFORMED ELASTIC AXIS AT G IS

$$\phi(g,t) = \frac{\partial}{\partial x}[\mu(x,g,t)] = \phi_{aft}\eta(t)$$

WHERE $\phi_{aft} = \frac{\partial h_g(x)}{\partial x_{aft}}$

Figure 2. Lateral Elastic Bending Geometry

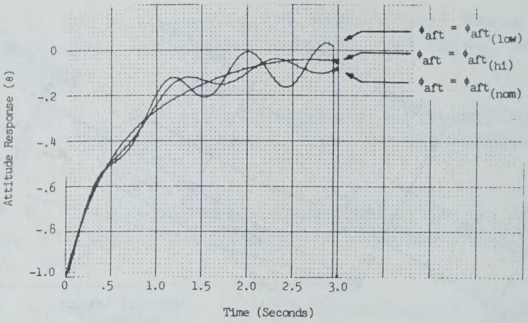


Figure 3a. Missile Attitude Response for S = 0

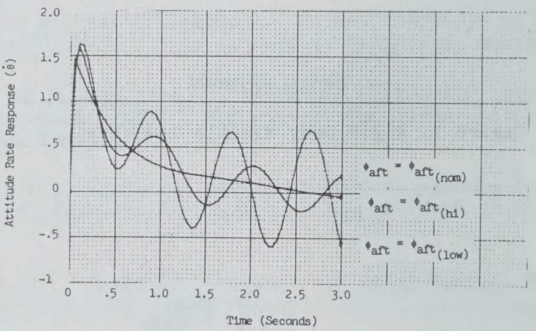


Figure 3b. Missile Attitude Rate Response for S = 0

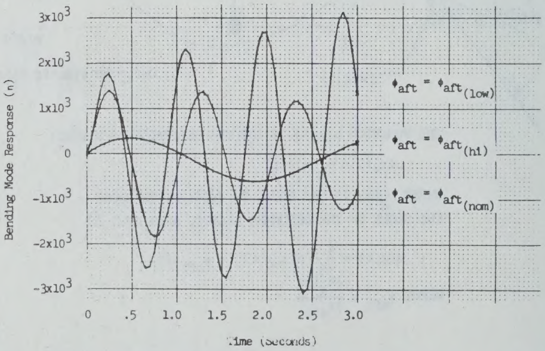


Figure 3c. Missile Bending Mode Response for S = 0

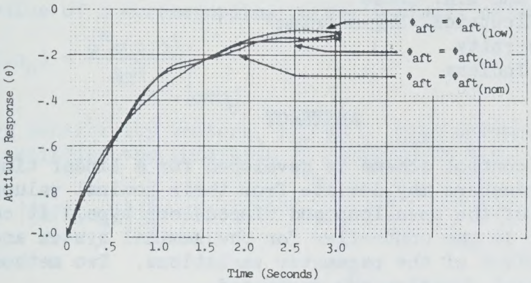


Figure 4a. Missile Attitude Response for $S = 1\text{x}10^{-16} I$

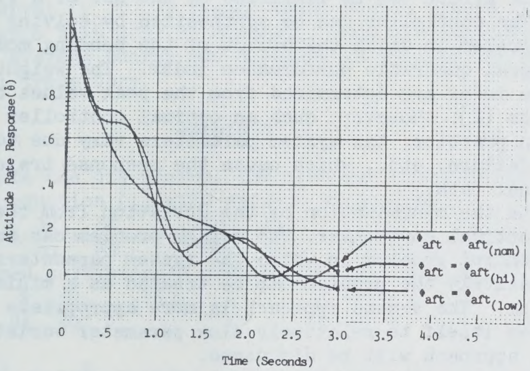


Figure 4b. Missile Attitude Rate Response for $S = 1\text{x}10^{-16} I$

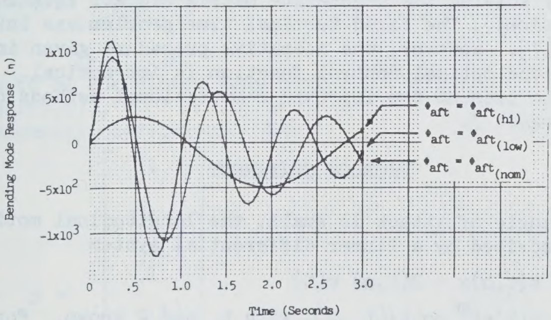


Figure 4c. Missile Bending Mode Response for $S = 1\text{x}10^{-16} I$

OPTIMAL CONTROLLER DESIGN FOR MINIMUM TRAJECTORY SENSITIVITY

J.Y.S. Luh and E.R. Cross
School of Electrical Engineering
Purdue University
Lafayette, Indiana

ABSTRACT

An optimal control scheme is developed for a linear time-varying system whose system parameters may deviate from their nominal values. The controller is a hybrid of the open-loop and closed-loop type. It consists of two parts, the first is the controller for the nominal system and the second minimizes the effect of the parameter variations. Two methods for computing the optimal control function are presented.

INTRODUCTION

The bending moments in a large launch booster often create undesirable disturbances. This effect can be minimized by the use of a linear feedback controller [1]. The controller can be synthesized by solving a conventional optimal control problem in which the effect of the bending moments is included in an integral quadratic performance index. The weighting matrices in the performance index are determined from the peak values of the bending moment coefficients [2]. However, such an optimal controller is not satisfactory because in practice, the system parameters vary due to the aging hardware, engine failure, etc., which cause the response trajectory to deviate from its nominal path.

To incorporate the minimization of the deviation from the nominal trajectory into the optimal controller, the design problem can be formulated as a stochastic control problem in which the system parameters are perturbed by noise. Alternatively the problem can be treated as a minimum trajectory sensitivity problem. The second approach is more appropriate if the dynamic system accommodates itself to relatively slow parameter variations. In this paper, the second approach will be discussed.

The problem of sensititivity in optimal control systems has been considered by various authors, e.g., Dorato [3], Pagurek [4], and Kahne [5]. Their work was based on first order approximations. D'Angelo, et al [6] investigated the trajectory deviation using a Taylor series approximation with infinite time. The constant gain factor in the closed loop controller was obtained by solving the degenerate matrix Riccati equation (solving algebraic equations). The fixed terminal time problem was investigated by Luh and Wozny [7]. However, the iterative procedure given in [7] for the computation of the optimal control function is impractical. The purpose of this paper is to present two improved computational methods over the algorithm given in Reference 7.

PROBLEM STATEMENT

When the angle of attack is small, the longitudinal motion of the booster can be described by a linear differential system

$$\dot{x} = A(t,\alpha)x + B(t,\alpha)\,u(t) \tag{1}$$

where $x(t)\epsilon R^n$, $u(t)\epsilon R^m$ on $t\epsilon[t_o,T]$, with t_o and T known. For simplicity, only one system parameter, α, is assumed to vary. It is also assumed that the state $x=x(t,\alpha)$ is sufficiently smooth in both t and α such that it can be approximated by a Taylor series

$$x(t,\alpha) \cong x(t,\alpha_o) + \sum_{k=1}^{M} s^{(k)}(t,\alpha_o)\,\frac{(\alpha-\alpha_o)^k}{k!}, \quad 1 \le M < \infty \tag{2}$$

where α_o is the value of α corresponding to the nominal trajectory $x(t,\alpha_o)$, and

$$s^{(k)}(t,\alpha_o) = \left.\frac{\partial^k x(t,\alpha)}{\partial\alpha^k}\right|_{\alpha=\alpha_o} \tag{3}$$

are the trajectory sensitivity vectors, $s^{(k)} \epsilon R^n$. Also assume that the control vector $u(t)$ satisfies the condition $\int_{t_o}^{T} ||u(t)||^2 dt < \infty$. Let

$$J(u) = ||x(T,\alpha_o)||_G^2 + \sum_{k=1}^{M} ||s^{(k)}(T,\alpha_o)||_{Q_K}^2 + \int_{t_o}^{T} [\,||x(t,\alpha_o)||_{W(t)}^2 + ||u(t)||_{U(t)}^2 + \sum_{k=1}^{M} ||s^{(k)}(t,\alpha_o)||_{S_k(t)}^2\,]dt \tag{4}$$

be the performance index where the matrices in the quadratic terms are symmetric and positive semi-definite except for $U(t)$ which is symmetric and positive definite. The problem is to determine a linear feedback controller on the time interval $[t_o,T]$, T fixed, which steers system (1) from $x(t_o,\alpha_o)$ to some $x(T,\alpha)$ such that $J(u)$ is minimal. Thus the controller is required to minimize the average energy and distance as well as the average deviation from the nominal trajectory.

ANALYSIS

To facilitate the discussion, the problem is reformulated as follows. Differentiating equation (1) with respect to α gives

$$\frac{\partial\dot{x}}{\partial\alpha} = A\,\frac{\partial x}{\partial\alpha} + \frac{\partial A}{\partial\alpha}\,x + B\,\frac{\partial u}{\partial\alpha} + \frac{\partial B}{\partial\alpha}\,u \tag{5}$$

Since x is assumed sufficiently smooth both in t and α,

$$\frac{\partial\dot{x}}{\partial\alpha} = \frac{d}{dt}\left(\frac{\partial x}{\partial\alpha}\right) = \dot{s}^{(1)}(t,\alpha) \tag{6}$$

By induction,

$$\frac{\partial^k\dot{x}}{\partial\alpha^k} = \frac{d}{dt}\left(\frac{\partial^k x}{\partial\alpha^k}\right) = \dot{s}^{(k)}(t,\alpha),\ k = 1,\ldots,M \tag{7}$$

where $s^{(k)}(t_o,\alpha)$ is taken as zero. Differentiating (5) and substituting into (7),

$$\dot{s}^{(k)}(t,\alpha) = \sum_{i=0}^{k}\binom{k}{i}\left[\frac{\partial^i A}{\partial\alpha^i}\,s^{(k-i)}(t,\alpha) + \frac{\partial^i B}{\partial\alpha^i}\,\frac{\partial^{k-i}u}{\partial\alpha^{k-i}}\right] \tag{8}$$

$$k=1,\ldots,M,$$

where u is taken to be the closed loop control, i.e.,

$$u = u(x(t,\alpha)), \tag{9}$$

and $\binom{k}{i}$ is the binomial coefficient. Let

$$y = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ s_1^{(1)} \\ \vdots \\ s_n^{(1)} \\ s_1^{(2)} \\ \vdots \\ s_n^{(M)} \end{bmatrix} \tag{10}$$

where $y \in R^{(M+1)n}$. The augmented system is obtained by combining (1), (8) and (10), i.e.,

$$\begin{cases} \dot y = \hat A(t,\alpha)y + \hat B(t,\alpha)u + \hat C(t,\alpha)v & (11) \\ y(t_o) = \begin{bmatrix} x(t_o) \\ 0 \end{bmatrix} & (12) \end{cases}$$

with

$$J(u) = ||y(T)||^2_{\hat G} + \int_{t_o}^{T} \left[||y(t)||^2_{\hat W} + ||u(t)||^2_{U(t)} \right] dt \qquad (13)$$

where the (M+1)n by (M+1)n matrix $\hat A(t,\alpha)$ in equation (11) is given by

$$\hat A(t,\alpha) = \begin{bmatrix} \binom{0}{0} A(t,\alpha) & 0 & 0 & \cdots & 0 \\ \binom{1}{1} \frac{\partial A}{\partial \alpha} & \binom{1}{0} A(t,\alpha) & 0 & \cdots & 0 \\ \binom{2}{2} \frac{\partial^2 A}{\partial \alpha^2} & \binom{2}{1} \frac{\partial A}{\partial \alpha} & \binom{2}{0} A(t,\alpha) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \binom{M}{M} \frac{\partial^M A}{\partial \alpha^M} & \binom{M}{M-1} \frac{\partial^{M-1} A}{\partial \alpha^{M-1}} & \binom{M}{M-2} \frac{\partial^{M-2} A}{\partial \alpha^{M-2}} & \cdots & \binom{M}{0} A(t,\alpha) \end{bmatrix} \qquad (14)$$

the (M+1)n by m matrix $\hat B(t,\alpha)$ by

$$\hat B(t,\alpha) = \begin{bmatrix} B(t,\alpha) \\ \partial B/\partial\alpha \\ \vdots \\ \partial^M B/\partial\alpha^M \end{bmatrix} \qquad (15)$$

the (M+1)n by Mm matrix $\hat C(t,\alpha)$ by

$$\hat C(t,\alpha) = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \binom{1}{0} B(t,\alpha) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \binom{M}{M-1} \frac{\partial^{M-1} B}{\partial \alpha^{M-1}} & \binom{M}{M-2} \frac{\partial^{M-2} B}{\partial \alpha^{M-2}} & \cdots & \binom{M}{0} B(t,\alpha) \end{bmatrix} \qquad (16)$$

the Mm-dimensional vector v(t) by

$$v' = \left[\partial u_1'/\partial\alpha \ \ldots \ \partial^M u_m'/\partial\alpha^M \right] \qquad (' \text{ denotes transpose}) \qquad (17)$$

and the (M+1)n by (M+1)n symmetric positive semi-definite matrices $\hat G$ and $\hat W(t)$ by

$$\hat G = \begin{bmatrix} G & & & 0 \\ & Q_1 & & \\ & & \ddots & \\ 0 & & & Q_M \end{bmatrix} (18), \quad \hat W(t) = \begin{bmatrix} W(t) & & & 0 \\ & S_1(t) & & \\ & & \ddots & \\ 0 & & & S_M(t) \end{bmatrix} \qquad (19)$$

It can be shown that the augmented system (11) has the known linear feedback optimal solution (Fig.1)

$$u^*(y,t) = -U(t)^{-1}\hat B(t,\alpha)'[R(t)y+h(t)] \qquad (20)$$

where $R(t) = R(t)'$ is (M+1)n by (M+1)n and satisfies the matrix Riccati equation

$$-\dot R(t) = R(t)\,\hat A(t,\alpha) + \hat A'(t,\alpha)\,R(t) - R(t)\,\hat B(t,\alpha)U^{-1}(t)\,\hat B(t,\alpha)'R(t) + \hat W(t) \qquad (21)$$

with

$$R(T) = \hat G \qquad (22)$$

and h(t) is an (M+1)n dimensional vector satisfying the linear differential equation

$$-\dot h(t) = [\hat A(t,\alpha)' - R(t)\hat B(t,\alpha)U(t)^{-1}\hat B(t,\alpha)']h(t) + R(t)\hat C(t,\alpha)v(t) \qquad (23)$$

with $\quad h(T) = 0. \qquad (24)$

When the optimal feed-back control u*(y,t) is implemented, the resulting system is obtained by combining (11) and (20):

$$\left\{\begin{array}{l} \dot{y}=[\hat{A}(t,\alpha)-\hat{B}(t,\alpha)U(t)^{-1}\hat{B}(t,\alpha)'R(t)]y-\hat{B}(t,\alpha)U(t)^{-1}\hat{B}(t,\alpha)'h(t)+\hat{C}(t,\alpha)v \qquad (25) \\ y(t_o)=\begin{bmatrix} x(t_o) \\ 0 \end{bmatrix} \qquad (26) \end{array}\right.$$

The (M+1)n dimensional vector $\hat{C}(t,\alpha)v$ in (25) can be expressed as follows. From (21), it is clear that R depends on α. Hence, differentiating (20) yields

$$\frac{\partial u}{\partial\alpha} = -U^{-1}(t)\left[\hat{B}'R\,\frac{\partial y}{\partial\alpha} + \left(\hat{B}'\,\frac{\partial R}{\partial\alpha} + \frac{\partial\hat{B}'}{\partial\alpha}\,R\right)y + \hat{B}'\,\frac{\partial h}{\partial\alpha} + \frac{\partial\hat{B}'}{\partial\alpha}\,h\right] \qquad (27)$$

where

$$\frac{\partial y}{\partial\alpha} = \frac{\partial}{\partial\alpha}\begin{bmatrix} x \\ s^{(1)} \\ \vdots \\ s^{(M)} \end{bmatrix} = \begin{bmatrix} s^{(1)} \\ \vdots \\ s^{(M)} \\ s^{(M+1)} \end{bmatrix} \cong \begin{bmatrix} s^{(1)} \\ \vdots \\ s^{(M)} \\ 0 \end{bmatrix}. \qquad (28)$$

Neglecting $s^{(M+1)}$ in the final form of (28) is justified by the assumptions made on the Taylor series approximation, i.e., that $s^{(k)}$ is negligible for $k > M$. The higher order terms follow by induction. Thus

$$\frac{\partial^k u}{\partial\alpha^k} = -\sum_{j=0}^{k}\left\{\sum_{i=0}^{j}\binom{k}{j}\binom{j}{i}U^{-1}(t)\left[\frac{\partial^i\hat{B}'}{\partial\alpha^i}\right]\left[\frac{\partial^{j-i}R}{\partial\alpha^{j-i}}\right]N^{(k-j)n}y + \binom{k}{j}U^{-1}\,\frac{\partial^j\hat{B}'}{\partial\alpha^j}\,\frac{\partial^{k-j}h}{\partial\alpha^{k-j}}\right\} \qquad (29)$$

where

$$N = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \qquad (30)$$

is an (M+1)n by (M+1)n nilpotent matrix. Since

$$\sum_{j=0}^{k}\binom{k}{j}\frac{\partial^j\hat{B}'}{\partial\alpha^j}\frac{\partial^{k-j}h}{\partial\alpha^{k-j}} = \sum_{j=0}^{k}\binom{k}{k-j}\frac{\partial^{k-j}\hat{B}'}{\partial\alpha^{k-j}}\,\frac{\partial^j h}{\partial\alpha^j} \qquad (31)$$

then

$$\sum_{k=1}^{\nu}\sum_{j=0}^{k}\binom{\nu}{\nu-k}\binom{k}{j}\frac{\partial^{\nu-k}B}{\partial\alpha^{\nu-k}}U^{-1}(t)\,\frac{\partial^j\hat{B}'}{\partial\alpha^j}\,\frac{\partial^{k-j}h}{\partial\alpha^{k-j}}$$

$$= [E_{\nu,0} \quad E_{\nu,1} \cdots E_{\nu,\nu} \quad 0 \cdots 0]z \qquad \nu=1,2,\ldots,M, \qquad (32)$$

where

$$E_{\nu,i} = \left[\sum_{k=1}^{\nu}\binom{\nu}{\nu-k}\binom{k}{k-i}\frac{\partial^{\nu-k}B}{\partial\alpha^{\nu-k}}U^{-1}(t)\,\frac{\partial^{k-i}\hat{B}'}{\partial\alpha^{k-i}}\right],\ i=0,1,\ldots,\nu. \qquad (33)$$

is an n by (M+1)n matrix and

$$z' = [h' \ \partial h'/\partial\alpha \ \ldots \ \partial^M h'/\partial\alpha^M] \qquad (34)$$

is an $(M+1)^2n$ dimensional vector. Consequently, combining (16), (17), (29), (32), (33) and (34) yields

$$\tilde{C}(t,\alpha)v = \hat{D}(t,\alpha)y + E(t,\alpha)z \qquad (35)$$

where

$$\hat{D}(t,\alpha) = -\begin{bmatrix} 0 \\ \displaystyle\sum_{k=1}^{1}\sum_{j=0}^{k}\sum_{i=0}^{j}\binom{1}{1-k}\binom{k}{j}\binom{j}{i}\frac{\partial^{1-k}B}{\partial\alpha^{1-k}}U^{-1}(t)\,\frac{\partial^i\hat{B}'}{\partial\alpha^i}\,\frac{\partial^{j-i}R}{\partial\alpha^{j-i}}\,N^{(k-j)n} \\ \vdots \\ \displaystyle\sum_{k=1}^{M}\sum_{j=0}^{k}\sum_{i=0}^{j}\binom{M}{M-k}\binom{k}{j}\binom{j}{i}\frac{\partial^{M-k}B}{\partial\alpha^{M-k}}U^{-1}(t)\,\frac{\partial^i\hat{B}'}{\partial\alpha^i}\,\frac{\partial^{j-i}R}{\partial\alpha^{j-i}}\,N^{(k-j)n} \end{bmatrix} \qquad (36)$$

is an (M+1)n by (M+1)n matrix and

$$E(t,\alpha) = -\begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ E_{1,0} & E_{1,1} & 0 & 0 & \cdots & 0 \\ E_{2,0} & E_{2,1} & E_{2,} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ E_{M,0} & E_{M,1} & E_{M,2} & E_{M,3} & \cdots & E_{M,M} \end{bmatrix} \tag{37}$$

is an $(M+1)n$ by $(M+1)^2 n$ matrix.

It remains to determine the expressions for $\partial^k \dot{h}/\partial\alpha^k$, $k=1,2,\ldots,M$. Combining (23) and (35) yields

$$-\dot{h}(t) = [\hat{A}(t,\alpha)' - R(t)\hat{B}(t,\alpha)U(t)^{-1}\hat{B}(t,\alpha)']h(t) + R(t)\,[\hat{D}(t,\alpha)y + E(t,\alpha)z] \tag{38}$$

Thus

$$-\frac{d}{dt}\left(\frac{\partial^k h}{\partial\alpha^k}\right) = \sum_{j=0}^{k} \binom{k}{j} \left\{ \left[\frac{\partial^{k-j}\hat{A}'}{\partial\alpha^{k-j}} - \sum_{i=0}^{k-j} \binom{k-j}{i} \frac{\partial^{k-j-i}R}{\partial\alpha^{k-j-i}} \sum_{\nu=0}^{i} \binom{i}{\nu} \frac{\partial^{i-\nu}\hat{B}}{\partial\alpha^{i-\nu}} U^{-1}(t) \frac{\partial^{\nu}\hat{B}'}{\partial\alpha^{\nu}} \right] \frac{\partial^j h}{\partial\alpha^j} + \frac{\partial^{k-j}R}{\partial\alpha^{k-j}} \sum_{\nu=0}^{j} \binom{j}{\nu} \left[\frac{\partial^{j-\nu}\hat{D}}{\partial\alpha^{j-\nu}} \frac{\partial^{\nu}y}{\partial\alpha^{\nu}} + \frac{\partial^{j-\nu}E}{\partial\alpha^{j-\nu}} \frac{\partial^{\nu}z}{\partial\alpha^{\nu}} \right] \right\} \tag{39}$$

or

$$-\frac{d}{dt}\left(\frac{\partial^k h}{\partial\alpha^k}\right) = \sum_{j=0}^{k} \binom{k}{j} \left[\frac{\partial^{k-j}\hat{A}'}{\partial\alpha^{k-j}} - \sum_{i=0}^{k-j} \sum_{\nu=0}^{i} \binom{k-j}{i} \binom{i}{\nu} \frac{\partial^{k-j-i}R}{\partial\alpha^{k-j-i}} \frac{\partial^{i-\nu}\hat{B}}{\partial\alpha^{i-\nu}} U^{-1}(t) \frac{\partial^{\nu}\hat{B}'}{\partial\alpha^{\nu}} \right] \frac{\partial^j h}{\partial\alpha^j} + \sum_{j=0}^{k} \sum_{\nu=0}^{j} \binom{k}{j}\binom{j}{\nu} \frac{\partial^{k-j}R}{\partial\alpha^{j-\nu}} \left[\frac{\partial^{j-\nu}\hat{D}}{\partial\alpha^{j-\nu}} N^{\nu n} y + \frac{\partial^{j-\nu}E}{\partial\alpha^{j-\nu}} N^{\nu(M+1)n} z \right] \tag{40}$$

$$k = 0, 1, \ldots, M.$$

Here $\partial^{\mu}h/\partial\alpha^{\mu}$ is neglected for $\mu \geq M+1$ for the same reason which justifies equation (28). Finally, let

$$\zeta = \begin{bmatrix} y \\ z \end{bmatrix} \tag{41}$$

be an $(M+1)(M+2)n$ dimensional vector, then combine (25), (26), (35), (38), (39) and (40) to yield a homogeneous differential system

$$\begin{cases} \dot{\zeta}(t) = H(t,\alpha)\,\zeta(t) \\ \zeta_i(t_o) = x_i(t_o), \quad i=1,2,\ldots,n, \\ \zeta_j(t_o) = 0, \quad j=n+1,n+2,\ldots,(M+1)n, \\ \zeta_k(T) = 0, \quad k=(M+1)n+1,\ldots,(M+1)(M+2)n. \end{cases} \tag{42}$$

where the $(M+1)(M+2)n$ square matrix

$$H(t,\alpha) = \left[\begin{array}{c|c} \hat{A}(t,\alpha) - \hat{B}(t,\alpha)\,U(t)^{-1}\,\hat{B}(t,\alpha)'R + \hat{D}(t,\alpha) & E(t,\alpha) \\ \hline -R(t)\,\hat{D}(t,\alpha) & -E(t,\alpha) \\ \hline -K(t,\alpha) & -P(t,\alpha) \end{array}\right] + \left[\begin{array}{c|c|c} & -\hat{B}(t,\alpha)\,U(t)^{-1}\,\hat{B}(t,\alpha) & 0_2 \\ \cline{2-3} 0_1 & \multicolumn{2}{c}{-L(t,\alpha)} \end{array}\right] \tag{43}$$

In (43), $\hat{A}$ is given in (14), $\hat{B}$ in (15), $\hat{D}$ in (36), E in (37); R is the solution of Riccati equation (21) and (22); $U(t)$ is symmetric positive definite and is associated with the performance index $J(u)$; 0_i, $i=1,2,$

are zero matrices with O_1 being (M+1)(M+2)n by (M+1)n and O_2 being (M+1)n by (M+1)Mn; and

$$K(t,\alpha) = \begin{bmatrix} \sum_{j=0}^{1} \sum_{\nu=0}^{j} \binom{1}{j}\binom{j}{\nu} \frac{\partial^{k-j} R}{\partial\alpha^{k-j}} \frac{\partial^{j-\nu} \hat{D}}{\partial\alpha^{j-\nu}} N^{\nu n} \\ \vdots \\ \sum_{j=0}^{M} \sum_{\nu=0}^{j} \binom{M}{j}\binom{j}{\nu} \frac{\partial^{k-j} R}{\partial\alpha^{k-j}} \frac{\partial^{j-\nu} \hat{D}}{\partial\alpha^{j-\nu}} N^{\nu n} \end{bmatrix} \tag{44}$$

is an (M+1)Mn by (M+1)n matrix,

$$P(t,\alpha) = \begin{bmatrix} \sum_{j=0}^{1} \sum_{\nu=0}^{j} \binom{1}{j}\binom{j}{\nu} \frac{\partial^{k-j} R}{\partial\alpha^{k-j}} \frac{\partial^{j-\nu} E}{\partial\alpha^{j-\nu}} N^{\nu(M+1)n} \\ \vdots \\ \sum_{j=0}^{M} \sum_{\nu=0}^{j} \binom{M}{j}\binom{j}{\nu} \frac{\partial^{k-j} R}{\partial\alpha^{k-j}} \frac{\partial^{j-\nu} E}{\partial\alpha^{j-\nu}} N^{\nu(M+1)n} \end{bmatrix} \tag{45}$$

is an (M+1)Mn by $(M+1)^2n$ matrix, and

$$L(t,\alpha) = \begin{bmatrix} L_{0,0} & 0 & 0 & \cdots & 0 \\ L_{1,0} & L_{1,1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ L_{M,0} & L_{M,1} & L_{M,2} & \cdots & L_{M,M} \end{bmatrix} \tag{46}$$

is an $(M+1)^2n$ by $(M+1)^2n$ matrix in which

$$L_{k,j} = \binom{k}{j}\left[\frac{\partial^{k-j}\hat{A}'}{\partial\alpha^{k-j}} - \sum_{i=0}^{k-j} \sum_{\nu=0}^{i} \binom{k-j}{\nu}\binom{i}{\nu} \frac{\partial^{k-j-i} R}{\partial\alpha^{k-j-i}} \frac{\partial^{i-\nu}\hat{B}}{\partial\alpha^{i-\nu}} U(t)^{-1} \frac{\partial^{\nu}\hat{B}'}{\partial\alpha^{\nu}}\right], \tag{47}$$

$$k,j=0,1,\ldots,M,$$

is an (M+1)n by (M+1)n matrix.

COMPUTATIONAL PROCEDURE

In this section two schemes are presented for computing the optimal feed-back control function $u^*(y,t)$ on $t \in [t_o,T]$. The first method is a straight forward solution of the time-varying homogeneous linear differential system (42). The second is an iterative scheme for solving equations (11), (17), (20), (23), and (40) recursively. Both schemes require the solution of Riccati equation (21) and the solutions of the matrix differential equation

$$-\frac{\partial^k \dot{R}}{\partial\alpha^k} = \sum_{i=0}^{k} \binom{k}{i}\left[\frac{\partial^{k-i} R}{\partial\alpha^{k-i}} \frac{\partial^{i}\hat{A}}{\partial\alpha^{i}} + \frac{\partial^{k-i}\hat{A}'}{\partial\alpha^{k-i}} \frac{\partial^{i} R}{\partial\alpha^{i}}\right] - \frac{\partial^k F}{\partial\alpha^k} \tag{48}$$

for k=1,2,...,M, where

$$F = R\hat{B}(t,\alpha)\, U(t)^{-1}\, \hat{B}(t,\alpha)'R \tag{49}$$

and

$$\frac{\partial^k F}{\partial\alpha^k} = \sum_{j=0}^{k} \sum_{i=0}^{k-j} \sum_{\nu=0}^{j} \binom{k}{j}\binom{k-j}{i}\binom{i}{\nu} \frac{\partial^{k-j-i} R}{\partial\alpha^{k-j-i}} \frac{\partial^{i}\hat{B}}{\partial\alpha^{i}} U(t)^{-1} \frac{\partial^{j-\nu}\hat{B}'}{\partial\alpha^{j-\nu}} \frac{\partial^{\nu} R}{\partial\alpha^{\nu}} \tag{50}$$

These solutions can be computed by numerically integrating (21) and (48) backwards from the terminal conditions $R(T)=\hat{G}$ and $\partial^k R(T)/\partial\alpha^k=0$, k=1,2,...,M, and should be stored for further usage.

Method 1

Let $\Phi(t,t_o;\alpha)$ be the state transition matrix for the homogeneous system (42)

$$\dot{\zeta} = H(t,\alpha)\,\zeta,$$

i.e., $\dot{\Phi}(t,t_o;\alpha) = H(t,\alpha)\Phi(t,t_o;\alpha)$ and $\Phi(t_o,t_o,\alpha) = I$.

Let

$$\Phi(t,t_o;\alpha)=\begin{bmatrix}\Phi_{11}(t,t_o;\alpha) & \Phi_{12}(t,t_o;\alpha)\\ \Phi_{21}(t,t_o;\alpha) & \Phi_{22}(t,t_o;\alpha)\end{bmatrix} \tag{51}$$

where Φ_{11} is $(M+1)n$ by $(M+1)n$, Φ_{12} is $(M+1)n$ by $(M+1)^2n$, Φ_{21} is $(M+1)^2n$ by $(M+1)n$, and Φ_{22} is $(M+1)^2n$ by $(M+1)^2n$. Then

$$\zeta(T) = \begin{bmatrix} y(T)\\ z(T)\end{bmatrix} = \begin{bmatrix}\Phi_{11}(T,t_o;\alpha) & \Phi_{12}(T,t_o;\alpha)\\ \Phi_{21}(T,t_o;\alpha) & \Phi_{22}(T,t_o;\alpha)\end{bmatrix}\begin{bmatrix} y(t_o)\\ z(t_o)\end{bmatrix} \tag{52}$$

Since $z(T) = 0$,

$$z(t_o) = -\,\Phi_{22}^{-1}(T,t_o;\alpha)\,\Phi_{21}(T,t_o;\alpha)\,y(t_o) \tag{53}$$

if the inverse $\Phi_{22}^{-1}(T,t_o;\alpha)$ exists. Thus the initial conditions of system (42) are completely known and $\zeta(t)$ can be computed by forward integration. Since $h_i(t) = \zeta_{(M+1)n+i}(t)$, $i=1,2,\ldots,(M+1)n$, $u^*(y,t)$ can be determined from equation (20).

Method 2

The optimal feed-back control function $u^*(y,t)$ is computed iteratively by first guessing $v(t)$ for all $t\in[t_o,T]$. Let $v^o(t)$ be the initial guess. Use $v^o(t)$ to compute $h^o(t)$ from (23) by backward integration. Now set $h=h^o$ and $v=v^o$ and integrate (25) forward to obtain $y^o(t)$. Consequently $u^o(t)$ can be computed from (20) and $J^o=J(u^o)$ from (13).

Differentiating (23) with respect to α yields

$$-\frac{d}{dt}\left(\frac{\partial^k h}{\partial\alpha^k}\right) = \sum_{i=0}^{k}\left[\binom{k}{i}\frac{\partial^{k-i}\Omega}{\partial\alpha^{k-i}}\frac{\partial^i h}{\partial\alpha^i} + \sum_{j=0}^{k-i}\binom{k-i}{j}\frac{\partial^{k-i-j}R}{\partial\alpha^{k-i-j}}\frac{\partial^j\hat{C}}{\partial\alpha^j}N^{im}v\right] \tag{54}$$

where $\Omega = \hat{A} - F$ with F defined in (49). Then

$$\frac{\partial^k\Omega}{\partial\alpha^k} = \frac{\partial^k\hat{A}}{\partial\alpha^k} - \frac{\partial^k F}{\partial\alpha^k} \tag{55}$$

where $\partial^k F/\partial\alpha^k$ is given in (50). Thus $\partial^k h^o/\partial\alpha^k$, $k=1,\ldots,M$, can be calculated by integrating (54) backward. Substituting $\partial^k h^o/\partial\alpha^k$ into (29) yields $\partial^k u^1/\partial\alpha^k$, and hence $v^1(t)$, an improvement on $v^o(t)$:

$$v^1(t) = \begin{bmatrix}\partial u^1/\partial\alpha\\ \vdots\\ \partial^M u^1/\partial\alpha^M\end{bmatrix} \tag{56}$$

The procedure is repeated until $J^{\mu+1} \cong J^{\mu}$ for some integer μ, i.e., repeated for a number of iterations until the performance index reaches a fixed quantity at which $u^{\mu+1}(t) \cong u^{\mu}(t)$ and $v^{\mu+1}(t) \cong v^{\mu}(t)$ on $t\in[t_o,T]$.

EXAMPLE

Method 2 was used to solve the following scalar system. Given $\dot{x}=-\alpha x+u$, $x(0)=10$, and $M=1$, minimize $J = \int_0^{10}(x^2+s^2+u^2)dt$ where $s(t)=\partial x/\partial\alpha$. α_o, the nominal value of α, was taken to be unity. The augmented system equations are

$$\begin{cases}\dot{x} = -\alpha x + u\\ \dot{s} = -x - \alpha s + \partial u/\partial\alpha.\end{cases}$$

The initial performance index is $J^o=80.167$. The iterative results are $J^1=60.497$, $J^2=59.681$, $J^3=60.767$, $J^4=60.029$, etc. The amplitude of the oscillation in the iterative values of J has decayed to 0.005% of its initial value by the 15-th iteration: (Fig. 2) ($v^o(t)$ is set equal to zero on $[t_o,T]$)

$J^{15}=60.078$, $J^{16}=60.077$, $J^{\mu}=60.078$, $\mu \geq 17$.

Thus the iterative procedure requires 15 integrations of two sets of two first order differential equations.

If Method 1 were used, on the other hand, it would require an integration of the 6 by 6 matrix differential equation of $\Phi(t,t_o;\alpha)$ once and then an integration of 6-dimensional vector differential equation of $\zeta(t)$ once. For this example, the computer storage and time requirements are about the same for the two methods. However, for large M and n, Method 1 becomes impractical because of the sizes of the H and Φ matrices. Since Method 2 involves the solution of uncoupled sets of equations, it is relatively easier to implement.

It is noted that in both methods the Riccati equation (21) need only be solved once. This is a major improvement over the algorithm described in [7] where the Riccati equation must be solved during each iteration.

REFERENCES

1. Graham, K.D., "Minimax Control of Large Launch Booster," Preprints of 1966 JACC, pp. 359-373.
2. NASA Marshall Space Flight Center, "Model Vehicle No. 2 for Advanced Control Studies."
3. Dorato, P., "On Sensitivity in Optimal Control Systems," IEEE Trans. on Automatic Control, Vol. 8, July 1963, pp. 256-257.
4. Pagurak, B., "Sensitivity of the Performance of Optimal Control Systems to Plant Parameter Variations," in Sensitivity Methods in Control Theory edited by L. Radanovic, Pergamon Press, 1966, pp. 383-395.
5. Kahne, S.J., "Low Sensitivity Design of Optimal Linear Control Systems," Proc. 4th Allerton Conference on Circuit and System Theory, University of Illinois, 1966, pp. 499-507.
6. D'Angelo, H., Moe, M.L. and Hendricks, T.C., "Trajectory Sensitivity of an Optimal Control System," ibid, pp. 489-506.
7. Luh, J.Y.S. and M.J. Wozny, "On the Trajectory Sensitivity of Linear Optimal Control Systems," Proc. of 1967 National Symposium of American Astronautical Society, Huntsville, Alabama, Vol. III, AAS 67-461 (GC-11), pp. 1-10.

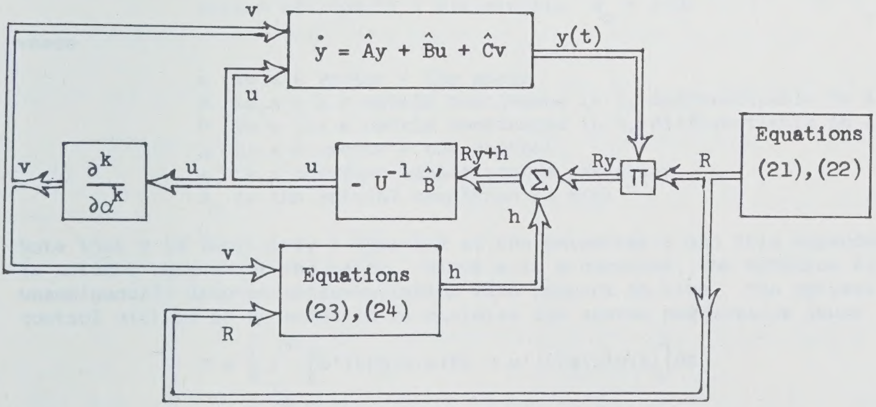


Figure 1. Linear Feedback Optimal Control for the Augmented System

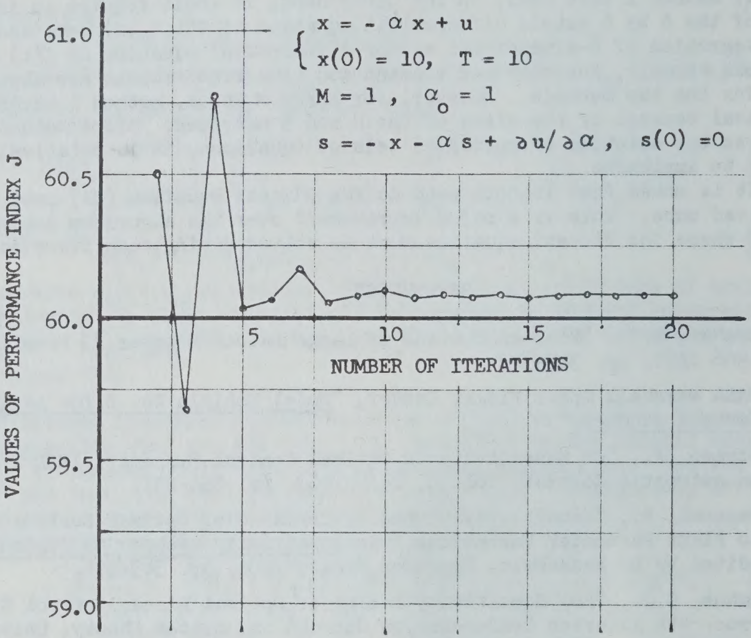


Figure 2. Iterative Values of the Performance Index for an Example

TIME DOMAIN SENSITIVITY COMPARISONS

G. B. LAMONT AND S. J. KAHNE
Honeywell Aerospace Division and University of Minnesota, Minneapolis, Minnesota.
Department of Electrical Engineering and Center for Control Sciences, University of Minnesota, Minneapolis, Minnesota.

ABSTRACT

It is well known that there are several ways to define a sensitivity function in the time domain. This paper reports the results of a study in which these definitions are compared. Controllers designed on the basis of these sensitivity functions are related to each other and conclusions drawn about their usefulness in control system design.

I. INTRODUCTION

In the determination of optimal control with respect to a quadratic performance criterion for linear systems it is usually assumed that system parameters are accurately known. However, since these parameters may deviate from their nominal values in the actual system to be controlled it is desirable to take these deviations into consideration in the selection of an optimal controller. A class of approaches discussed recently has accounted for the parameter changes by designing controllers which, in some sense, are less sensitive to parameter deviations. Certain approximations have been made in these approaches and the precise effect of these approximations has not been clear. This paper discusses several suggested approaches and compares them with each other, illustrating, by means of an example the differences between resulting system designs.

Since each approach has been derived elsewhere, derivations are not included in the present paper. References to the derivations are given.

II. QUADRATIC PERFORMANCE INDEX LINEAR OPTIMAL CONTROL PROBLEMS

The general system to be considered is the linear plant defined by the state vector equations

$$\dot{x}(t) = A(t,a)x(t) + B(t,a)u(t), \quad x_o = x(0) \tag{L}$$

where

x is a n vector - the state
A is a n x n matrix continuous in t, differentiable in a
B is a n x m matrix continuous in t, differentiable in a
u is a m vector - the control
a is a constant scalar system parameter
x_o is the initial condition on x(t)

Note that x is implicitly a function of the parameter a and this dependence is assumed throughout the paper. Since a is a constant, the notation $\dot{x}(t)$ unambiguously denotes differentiation with respect to time. The optimal control u(t) is to be selected to minimize the system performance index

$$J = \frac{1}{2} \int_{t_o}^{T} \left[x'(t)Q(t)x(t) + u'(t)R(t)u(t)\right]dt$$

where

Q is a symmetric semi-definite n x n matrix
R is a symmetric positive definite m x m matrix
Both Q(t) and R(t) are twice differentiable in t

The calculus of variations approach as defined by Kalman(1) is utilized to find the optimal control for the system. Thus,

$$u = - R^{-1}(t)B'(t,a)K(t,a)x(t)$$

where K(t) satisfies the vector Riccati equation:

$$- \dot{K}(t,a) = Q(t) + K(t,a)A(t,a) + A'(t,a)K(t,a) - K(t,a)B'(t,a)R^{-1}(t)B\ (t,a)K(t,a)$$

III. SENSITIVITY RESULTS FOR LINEAR OPTIMAL CONTROL PROBLEMS

If a sensitivity vector is defined as $s(t,a) = \frac{\partial x}{\partial a}(t,a)$, the differential equation in s becomes, from the system equation,

$$\dot{s}(t,a) = \frac{\partial A(t,a)}{\partial a}\, x(t) + A(t,a)s(t,a) + \frac{\partial B(t,a)}{\partial a}\, u(t,a) + B(t,a)\frac{\partial u(t,a)}{\partial a}\ , \quad s(0,a)=0$$

The sensitivity at t = 0 is zero since it is assumed that the parameter variations do not affect s at t = 0.

At this point, the published approaches differ. Kahne(2) assumes that the effect of the $\frac{\partial u(t,u)}{\partial a}$ term in the sensitivity equation is negligible and that B is not a function of a. Thus, the total system equations in this case are

$$\dot{x} = A(t,a)x(t) + B(t,a)u(t),\ x_o$$

$$\dot{s}(t,a) = \frac{\partial A}{\partial a}\, x(t) + As(t,a),\quad s(0,a) = 0$$

A performance index including sensitivity is suggested, namely,

$$J = \frac{1}{2}\int_{t_o}^{T} \Big[x'(t)Q(t)x(t) + u'(t)R(t)u(t) + s'(t)N(t)s(t)\Big]dt$$

where the origin is the desired value of x(t) at t = T. N(t) is a symmetric positive semi-definite n x n matrix and at least twice differentiable in t. The optimal control is

$$u(t) = - R^{-1}(t)B'(t,a)K_{11}(t,a)x(t) - R^{-1}(t)B'(t,a)K_{12}(t,a)s(t,a)$$

where $K_{11}(t,a)$ and $K_{12}(t,a)$ are defined by

$$K(t,a) = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}$$

and K(t) satisfies the Riccati equation

$$K(t,a) = \begin{bmatrix} Q & O \\ O & N \end{bmatrix} + K \begin{bmatrix} A & 0 \\ \frac{\partial A}{\partial a} & A \end{bmatrix} + \begin{bmatrix} A' & (\frac{\partial A}{\partial a})' \\ 0 & A' \end{bmatrix} K - K \begin{bmatrix} BR'^{-1}B & 0 \\ 0 & 0 \end{bmatrix} K$$

In this paper, the sensitivity vector was defined to be

$$s(t,a) = \frac{\partial x(t,a)}{\partial a}\Bigg|_{a_o = a \text{ nominal}}$$

Questions about such a definition and its governing differential equation are discussed in Section IV.

Dompe and Dorf$^{(5)}$ show that although an optimal controller does not result using the controller which is derived in Kahne's paper; the approach does indeed produce a system with very desirable sensitivity properties.

D'Angelo and others$^{(3)}$ have developed a sensitivity differential equation

$$\frac{\partial^i \dot{s}(t,a)}{\partial a^i} = \sum_{r=0}^{i+1} \binom{i+1}{r} \frac{\partial^r A}{\partial a^r} \frac{\partial^{i+1-r} x(t,a)}{\partial a^{i+1-r}} -$$

$$\sum_{r'=0}^{i+1} \sum_{v=0}^{L} \sum_{q=0}^{i+1-r'} \sum_{m=0}^{q} \binom{i+1}{r'} \binom{i+1-r}{q} \binom{q}{m} \frac{\partial^{r'} B}{\partial a^{r'}} R^{-1} \frac{\partial^{q-m} B}{\partial a^{q-m}} \frac{\partial^m K_{ov}}{\partial a^m} \frac{\partial^{v+i+1-r'-q} x(t,a)}{\partial a^{v+i+1-r'-q}}$$

$$i = 0,1,2,\ldots L,$$

$$\frac{\partial^i s(0,a)}{\partial a^i} = 0$$

by assuming that

$$u(t,a) = \sum_{v=0}^{n} - R^{-1}(t)B'(t,a)K_{ov}(t)\frac{\partial^v x(t,a)}{\partial a^v}$$

and $s(t,a) = \frac{\partial x(t,a)}{\partial a}$. It should be noted that they also assumed $\frac{\partial^m K_{ov}}{\partial a^m} = 0$ for $m \geq 1$ where the K_{ov}'s are symmetric partitioned matrices of K, with dimension n x n. Using the original plant equation, the sensitivity equation and the performance index

$$J = \frac{1}{2} \int_{t_o}^{T} [u'Ru + x'Qx + \hat{s}'M\hat{s}]dt$$

where

$$\hat{s} = \left[s', \frac{\partial s'}{\partial a}, \ldots \frac{\partial s'}{\partial a^L} \right]'$$

and M is a nL x Ln symmetric positive semi-definite matrix, the optimal control is found to be

$$u(t) = - R^{-1}(t)B'(t)K(t,a) \begin{bmatrix} x(t) \\ \hat{s}(t,a) \end{bmatrix}$$

K(t,a) satisfies

$$- \dot{K}(t,a) = \begin{bmatrix} Q(t) & 0 \\ 0 & M(t) \end{bmatrix} \begin{matrix} + K(t,a)F(K,t,a) + F'(K,t,a)K(t,a) \\ - K(t,a)B'(t,a)R^{-1}B\ (t,a)K(t,a) \end{matrix}$$

where

$$F_{kj} = \binom{k}{k-j} \frac{\partial^{k-j} A}{\partial a^{k-j}} +$$

$$\sum_{r,=0}^{k} \sum_{v=0}^{L} \sum_{m=0}^{v+k-r'-j} \binom{k}{r'} \binom{j}{v+k-r,-j} \binom{v+k-r,-j}{m} \frac{\partial^{r'} B}{\partial a^{r'}} R^{-1} \frac{\partial B^{v+k-r'-j}}{\partial a^{v+k-r'-j}} \frac{\partial^m k_{ov}}{\partial a^m} .$$

Note that the K(t,a) equation is not a Riccati equation since F is a function of K. The examples presented in D'Angelo's paper assume that all the terms in the sensitivity equation are evaluated at a_o (a nominal) except the output of the plant $x(t)$. Again, the questions arising from this definition are discussed in the next section.

IV. PROBLEM AREAS

The methods described in Section III differed in their definition of sensitivity. This resulted in different differential equations for sensitivity. The linear plant to be controlled is described by the system of linear differential equations (L).

Three sensitivity definitions including those mentioned above are now considered.

Approach I) $s = \frac{\partial x}{\partial a}$

$$\dot{s} = \frac{\partial A}{\partial a} x + As + \frac{\partial B}{\partial a} u + B \frac{\partial u}{\partial a}, \quad s(0,a) = 0$$

Note that in this approach the variations in a must be known to calculate sensitivity performance.

Approach II) $s = \frac{\partial x}{\partial a}$

$$\dot{s} = \frac{\partial A}{\partial a}\Big|_{a_o} x + A\Big|_{a_o} s + \frac{\partial B}{\partial a}\Big|_{a_o} u + B\Big|_{a_o} \frac{\partial u}{\partial a}, \quad s(0,a) = 0$$

In this case, it is assumed that the variations in a are not known except as reflected in the state x. Thus, only certain terms of the sensitivity equation are evaluated at a_o. Here the state $x(t,a)$ is used directly but the other coefficients, not directly available from the system are evaluated at a_o.

Approach III) $s = \frac{\partial x}{\partial a}$

$$\dot{s} = \frac{\partial A}{\partial a}\Big|_{a_o} x\Big|_{a_o} + A\Big|_{a_o} s + \frac{\partial B}{\partial a}\Big|_{a_o} u\Big|_{a_o} + B\Big|_{a_o} \frac{\partial u}{\partial a}\Big|_{a_o},$$

$$s(0,a_o) = 0$$

This is an open loop approach in which only the nominal value of a is known, and the sensitivity function is based on the nominal behavior of the system.

In addition to the three approaches above, the effect of neglecting $\frac{\partial u}{\partial a}$ is included.

V. EXTENSIONS

In this section the definitions of sensitivity are summarized for compactness including the optimal feedback equation and first order sensitivity.

Approach I)

$$\dot{x} = A(t,a)x + B(t,a)u, \quad x(0) = x_o$$

$$\dot{s} = \frac{\partial A}{\partial a} x + As + \frac{\partial B}{\partial a} u + B \frac{\partial u}{\partial a}, \quad s(0,a) = 0$$

$$J = \frac{1}{2} \int_{t_o}^{T} (u'Ru + x'Qx + s'Ms)dt$$

$$u = -R^{-1}B'(K_1 x + K_2 s)$$

where

$$\begin{bmatrix} \dot{K}_1 & \dot{K}_2 \\ \dot{K}_2 & \dot{K}_3 \end{bmatrix} = \dot{K} = F'K + KF + N - KB'R^{-1}BK$$

$$N = \begin{bmatrix} Q & 0 \\ 0 & M \end{bmatrix}$$

$$F_{k_j} = \binom{k}{k-j} \frac{\partial^{k-j} A(t,a)}{\partial a^{k-j}} +$$

$$\sum_{r=0}^{k} \sum_{v=0}^{L} \sum_{m=0}^{v+k-r-j} \binom{k}{r} \binom{j}{v+k-r'-j} \binom{v+k-r-j}{m} \frac{\partial^r B(t,a)}{\partial a^r} \cdot$$

$$R^{-1} \frac{\partial G^{v+k-r-j}}{\partial a^{v+k-r-j}} K_{ov}$$

If the u terms were not included in the sensitivity equation

$$F = \begin{bmatrix} A & 0 \\ \frac{\partial A}{\partial a} & A \end{bmatrix}$$

The feedback gains, K_i, are evaluated at a_o.

Approach II)

$$\dot{x} = A(t,a)x + B(t,a)u, \quad x(0) = x_o$$

$$\dot{s} = \frac{\partial A}{\partial a}(t,a_o)x + A(t,a_o)s + \frac{\partial B}{\partial a}(t,a_o)u + B(t,a_o) \frac{\partial u}{\partial a}, \quad s(0,a) = 0$$

$$J = \frac{1}{2} \int_{t_o}^{T} (u'Ru + x'Qx + s'Ms)dt; u = - R^{-1}B'(K_1x + K_2s)$$

where

$$\begin{bmatrix} \dot{K}_1 & \dot{K}_2 \\ \dot{K}_2 & \dot{K}_3 \end{bmatrix} = \dot{K} = F'K + KF + N - KB'R^{-1}BK$$

F and N are the same as in Approach I.

Approach III)

$$\dot{x} = A(t,a)x + B(t,a)u, \quad x(0) = x_o$$

$$\dot{s} = \frac{\partial A}{\partial a}(t,a_o)x(t,a_o)+A(t,a)s+ \frac{\partial B}{\partial a}(t,a_o)u(t,a_o)+B(t,a_o)\frac{\partial u}{\partial a}(t,a_o),s(0,a_o) = 0$$

$$J = \frac{1}{2} \int_{t_o}^{T} (u'Ru+x'Qx+s'Ns)dt$$

$$u = - R^{-1}B'(K_1x+K_2s)$$

$$\begin{bmatrix} \dot{K}_1 & \dot{K}_2 \\ \dot{K}_2 & \dot{K}_3 \end{bmatrix} = \dot{K} = F'K + KF + N - KB^1R^{-1}BK$$

F and N are again evaluated at a_o for the generation of the feedback gains, K_i. This approach requires that x be evaluated at a_o.

VI. EXAMPLE

The example consists of a first order plant. This example shows system response for various definitions of sensitivity as present in Section V. The system parameter is varied by 10% and a step input response is presented for each approach. Let R = Q = M = 1. The first order plant is described by

$$\dot{x} = a^2 x + u \; ; \quad x(0) = 0$$

which is unstable. With $\frac{\partial x}{\partial a} = s$

$$\dot{s} = 2ax + a^2 s + \frac{\partial u}{\partial a}, \quad s(0,a) = 0$$

$$J = \frac{1}{2} \int_0^{\infty} (u^2 + x^2 + s^2)\,dt$$

and

$$u = -(K_1 x + K_2 s)$$

where

$$2a_o^2 K_1 + 1 - K_1^2 + 4a_o K_2 = 0$$

$$2a_o^2 K_2 + 2a_o K_3 - 2K_1 K_2 = 0$$

$$2a_o^2 K_3 + 1 - K_2^2 - 2K_1 K_3 = 0$$

and $a_o = 1$.

Figure 1 presents the system without sensitivity control. The response of the system is shown in Figure 2 for a step input.

Figure 3 is a system block diagram for the inclusion of the first order sensitivity in the feedback loop. Sensitivity Approach I is utilized and the $\frac{\partial u}{\partial a}$ turns are neglected. Figure 4 presents the response of this system to a step input. The gain equations for this condition are

$$2a^2 K_1 + 1 - K_1^2 + 4aK_2 = 0$$

$$2a^2 K_2 + 2aK_3 - K_1 K_2 = 0$$

$$2a^2 K_3 + 1 - K_2^2 = 0$$

Figure 5 is the system block diagram using first order sensitivity control with the $\frac{\partial u}{\partial a}$ terms included. Figure 6 and 7 present the system responses for approaches I and II for this configuration.

Figure 8 is a representation of Approach III applied to the example. Figure 9 is the system response due to a step input.

VII. CONCLUSION

As shown in the various responses, the use of sensitivity feedback reduces plant excursions due to plant parameter variations. However, for the system where the $\frac{\partial u}{\partial a}$ turn was not included this is not true (see Figure 4). Thus, this term should be included for desired sensitivity performance in this example. The effect of neglecting this term will vary depending upon the particular plant.(4) Reference 5 shows that neglecting this term produces a non-optimal, but low sensitivity system.

Note that the system responses for the three approaches are nearly identical. This implies that all three definitions produce similar results and that Approach II, being the simplest configuration, is as good as the others. This result is expected since the system transfer functions are nearly identical when factored. This would seem to be the case for application to higher order systems as well.

VIII. FURTHER INVESTIGATIONS

Additional effort is required to fully evaluate the previous sensitivity definitions. For example, a detailed proof of the existence of a solution to the feedback gain equation, K, should be developed. Also, the question arises, should the term $\partial^m K_{cv} / \partial \alpha^m$ be included in the development? Its inclusion would mean solving a partial differential equation to find the feedback gain matrix K. A comparison between this development and the frequency domain sensitivity minimization approach of Horowitz$^{(6)}$ would be useful.

REFERENCES

1. Kalman, R. E., "Contributions to the Theory of Optimal Control," Bol. Soc. Math. Mexico, pp. 102-119, 1960.

2. Kahne, S. J., "Low sensitivity Design of Optimal Linear Control Systems," Allerton Conference, 1966.

3. D'Angelo, H., Hendricks, T. C., Moe, M. L., "Trajectory Sensitivity of an Optimal Control System," Allerton Conference, 1966.

4. Lamont, G. B., "Design of Minimum Sensitivity of Linear Optimal Control Systems," Master Paper, University of Minnesota, 1967.

5. Dompe, R. J., Dorf, A. C., "The Design of Optimal Low Sensitivity Linear Control Systems," Fifth Allerton Conference, 1967.

6. Horowitz, Synthesis of Feedback Systems, Academic Press, 1963.

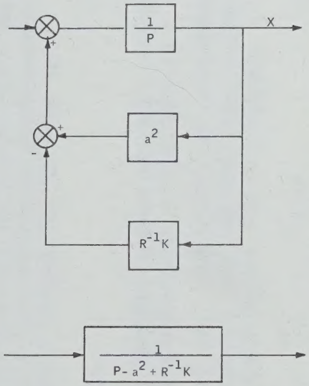

X
+
+
-
$\frac{1}{P}$
a^2
$R^{-1}K$
$\frac{1}{P - a^2 + R^{-1}K}$

Fig. 1. System 1, no sensitivity feedback

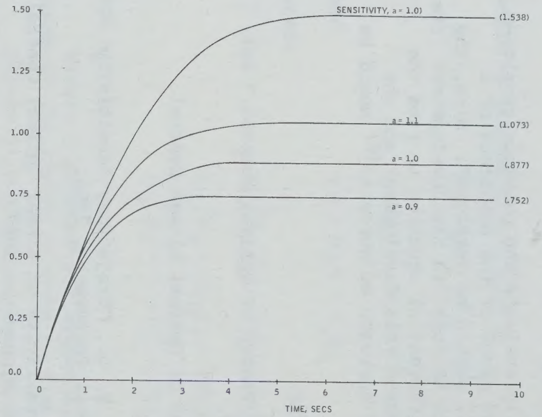


Fig. 2. System 1 response, $K_1 = 2.14$

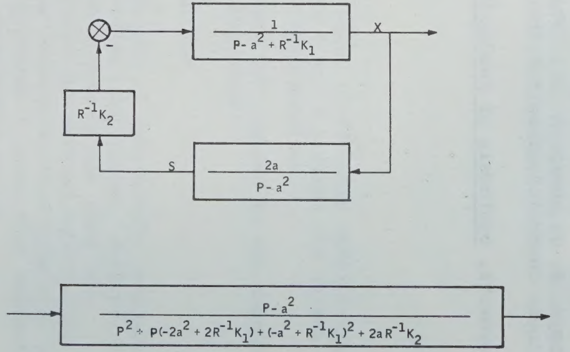

X
-
S
$\frac{1}{P - a^2 + R^{-1}K_1}$
$R^{-1}K_2$
$\frac{2a}{P - a^2}$
$\frac{P - a^2}{P^2 + P(-2a^2 + 2R^{-1}K_1) + (-a^2 + R^{-1}K_1)^2 + 2aR^{-1}K_2}$

Fig. 3. System 2, sensitivity fdbk, no $\frac{\partial u}{\partial a}$ terms

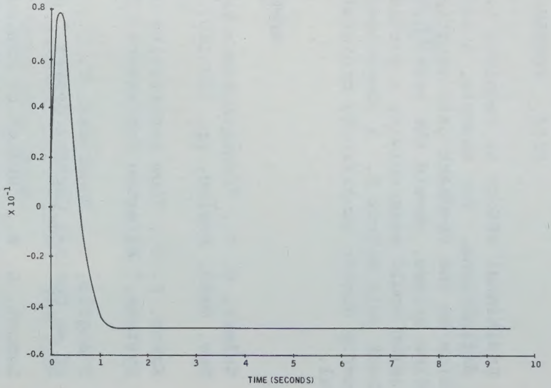


Fig. 4. System 2 response, $K_1 = 4.81$, $K_2 = 3.12$

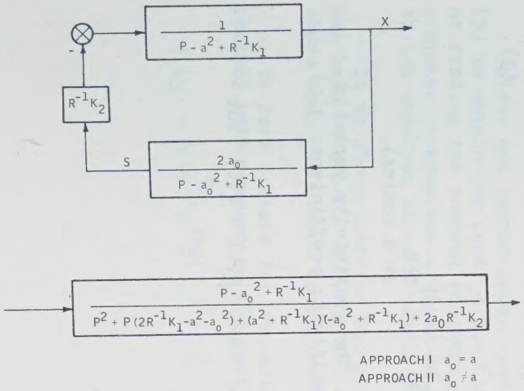

1
P - a2 + R-1K1
X
-
R-1K2
S
2 a0
P - a0 2 + R-1K1
P - a0 2 + R-1K1
P2 + P (2R-1K1-a2-a0 2) + (a2 + R-1K1)(-a0 2 + R-1K1) + 2a0R-1K2
APPROACH I a0 = a
APPROACH II a0 ≠ a

Fig. 5. System 3

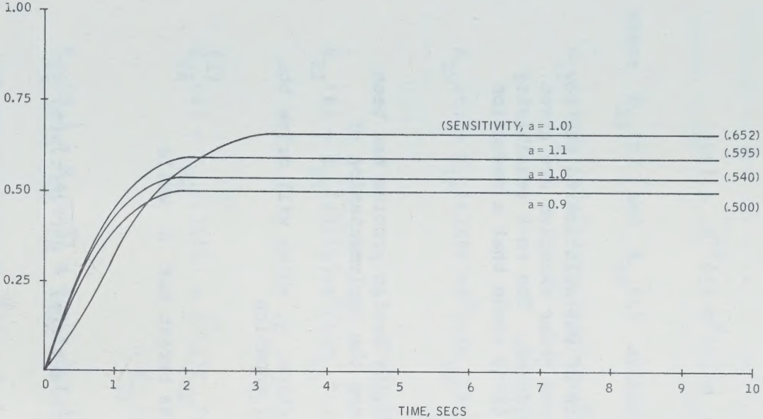


Fig. 6. System 3 response, Approach I, $K_1 = 2.65$, $K_2 = .18$

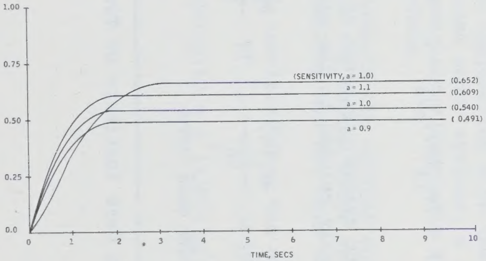


Fig. 7. System 3 response, Approach II, $K_1 = 2.65$, $K_2 = .18$

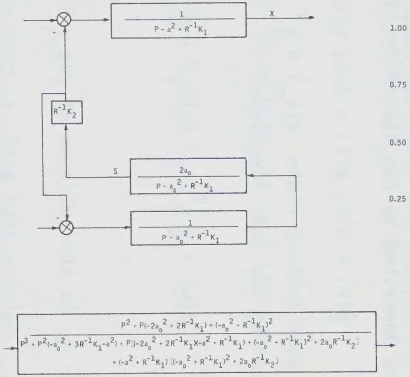

1
P - a2 + R-1K1
X
-
R-1K2
S
2a0
P - a0 2 + R-1K1
-
1
P - a0 2 + R-1K1
P2 + P(-2a0 2 + 2R-1K1) + (-a0 2 + R-1K1)2
P3 + P2(-a0 2 + 3R-1K1-a2) + P[(-2a0 2 + 2R-1K1)(-a2 + R-1K1) + (-a0 2 + R-1K1)2 + 2a0R-1K2]
+ (-a2 + R-1K1) [(-a0 2 + R-1K1)2 + 2a0R-1K2]

Fig. 8. System 4, Approach III

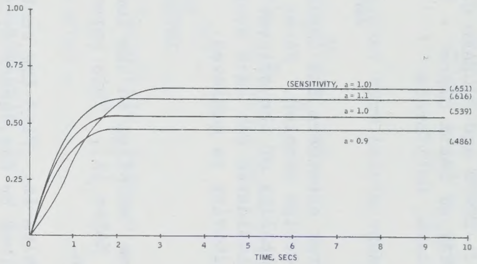


Fig. 9. System 4 response, $K_1 = 2.65$, $K_2 = .18$

THE DESIGN OF OPTIMAL LOW SENSITIVITY LINEAR CONTROL SYSTEMS

ROBERT J. DOMPE* and RICHARD C. DORF
University of Santa Clara
Santa Clara, California

ABSTRACT

This paper considers the implementation of one solution to the low sensitivity design problem. First and second order examples are given and the optimallity of the systems is considered. The root sensitivity measure is considered for each design and it is shown that a reduction in this sensitivity is achieved.

INTRODUCTION

An optimal solution to the low sensitivity design problem has been presented by Kahne [1]. This paper considers the implementation of this solution.

The problem is: Find the control function, $\underline{u}$, which will drive the system described by the vector differential equation

$$\dot{\underline{x}}(t) = A(t)\ \underline{x}(t) + B(t)\ \underline{u}(t) \qquad (1)$$

from some initial state $\underline{x}(t_o) = \underline{x}^o$ to some target set S while minimizing the cost functional

$$J(\underline{u}) = \frac{1}{2}\,\underline{x}^T(t_f)\ F\underline{x}(t_f) + \frac{1}{2}\int_{t_o}^{t_f}(\underline{x}^T(t)Q(t)\underline{x}(t) + \underline{u}^T(t)R(t)\underline{u}(t) + \underline{s}^T(t)\ \Sigma\ (t)\ \underline{s}(t))dt \qquad (2)$$

where $\underline{s}$ is the solution of the vector differential equation

$$\dot{\underline{s}}(t) = A(t)\ \underline{s}(t) + C(t)\ \underline{x}(t)\ , \qquad (3)$$

$$A(t) = [a_{ij}(t)]_{n\times n}\quad ,\quad C(t) = \left[\frac{\partial a_{ij}(t)}{\partial\alpha}\right]_{n\times n}\ \ \alpha = \alpha \text{ nominal}$$

and α is the system parameter of interest. The sensitivity variable $\underline{s}(t)$ is closely related to the vector of sensitivity coefficients $\underline{v}(t)$, where

$$v_i(t) = \frac{\partial x_i(t)}{\partial\alpha}\ .\ \text{If}\ \frac{\partial\underline{u}(t)}{\partial\alpha} \equiv 0,\ \text{then}\ \underline{s}(t) = \underline{v}(t)\ \text{but}$$

that is not the case here.

---

[*] Presently at Case Institute of Technology

The optimal control is

$$\underline{u}(t) = -R^{-1}(t)\ B^T(t)\ K_{11}(t)\ \underline{x}(t) - R^{-1}(t)\ B^T(t)\ K_{12}(t)\ \underline{s}(t) \qquad (4)$$

where $K_{11}(t)$ and $K_{12}(t)$ satisfy the Riccati type matrix equations:

$$\dot{K}_{11}(t)+K_{11}(t)A(t)+K_{12}(t)C(t)+A^T(t)K_{11}(t)+C^T(t)K_{21}(t)-$$

$$- K_{11}(t)B(t)R^{-1}(t)B^T(t)K_{11}(t) + Q(t) = 0$$

$$\dot{K}_{12}(t) + K_{12}(t)A(t)+A^T(t)K_{12}(t)+C^T(t)K_{22}(t) -$$

$$- K_{11}(t)B(t)R^{-1}(t)B^T(t)K_{12}(t) = 0$$

$$\dot{K}_{21}(t) + K_{21}(t)A(t)+A^T(t)K_{21}(t) + K_{22}(t)C(t) -$$

$$- K_{21}(t)B(t)R^{-1}(t)B^T(t)K_{11}(t) = 0$$

$$\dot{K}_{22}(t) + K_{22}(t)A(t) + A^T(t)K_{22}(t) -$$

$$- K_{21}(t)B(t)R^{-1}(t)B^T(t)K_{12}(t) + \sum(t) = 0$$

$$K_{12}(t_f) = K_{21}(t_f) = K_{22}(t_f) = 0, \quad K_{11}(t_f) = F . \qquad (5)$$

Where $K_{11}(t) = K_{11}^T(t)$,

$K_{22}(t) = K_{22}^T(t)$and $K_{21}(t) = K_{12}^T(t)$ [1].

First and second order systems were examined using equations (4) and (5) to obtain the control function $\underline{u}$. In conjunction with the problem of finding the control function for the second order systems, a digital computer program was written to solve ten non-linear algebraic equations in ten unknowns using a simple iterative scheme.

It is shown that for the first order example the optimal control has been obtained. This is not the case for the second order systems where there is reason to believe that optimallity has not been achieved.

To form a basis for comparison of root sensitivities, the optimal control with respect to the cost functional

$$J_1(\underline{u}) = \frac{1}{2}\ \underline{x}^T(t_f)F\underline{x}(t_f) + \frac{1}{2} \int_{t_o}^{t_f} (\underline{x}^T(t)Q(t)\underline{x}(t) + \underline{u}^T(t)R(t)\underline{u}(t))dt \qquad (6)$$

is found for each of the example systems. The root sensitivities of the corresponding systems are then compared. In the examples studied as much as a three-fold improvement in root sensitivity is realized.

CALCULATION OF OPTIMAL CONTROLS

First Order System

The first order system studied is the system described by the differential equation

$$\dot{x}(t) = - ax(t) + u(t)$$

where a nominal = 1, with $t_o = 0$, $t_f = \infty$, $F = 0$ and $Q = R = 1$. The solution for the optimal control when the cost functional is (6) is given in Athans and Falb [2]. Using that solution, the optimal control is $u^*(t) = -Kx(t)$ where $-Ka - aK - K^2 + 1 = 0$ and K is positive. Thus, $K = -1 + \sqrt{2}$ and $u^*(t) = (1 - \sqrt{2})\, x(t)$. This will be called the "standard" optimal control.

When the cost functional is (2), with $t_o = 0$, $t_f = \infty$, $F = 0$, and $Q = R = \Sigma = 1$, the optimal control is $u^*(t) = -K_{11}x(t) - K_{12}s(t)$ where

$$-2K_{11} - 2K_{12} - K_{11}^2 + 1 = 0$$

$$-2K_{12} - K_{22} - K_{11}K_{12} = 0$$

$$-2K_{22} - K_{12}^2 + 1 = 0 \qquad (7)$$

and $\dot{s}(t) = -as(t) - x(t)$, $s(o) = 0$. Solving (7) for K_{11} yields $K_{11}^4 + 8K_{11}^3 + 18K_{11}^2 + 8K_{11} - 11 = 0$. The roots of this equation are $-1.999999 \pm j\, 0.681338$, 0.545082, and -4.545083. It is shown elsewhere that the matrix

$$\begin{bmatrix} K_{11} & K_{12} \\ K_{12}^T & K_{22} \end{bmatrix}$$

must be positive definite [3]. Thus, $K_{11} = 0.545082$ and $K_{12} = -0.193589$. Therefore,

$$u^*(t) = -0.545082\, x(t) + 0.193589\, s(t) .$$

This will be called the "sensitivity" optimal control.

To check on the optimallity of these controls the system was simulated by implementing equations (1), (2), (3), and (6) on an analog computer. Control functions of the form $u(t) = kx(t)$ and $u(t) = -k_1x(t) - k_2s(t)$ were generated as shown in Fig. 1. In each case the respective cost functionals were located at local minimums when the calculated optimal controls were applied.

Second Order Systems

The second order system considered is the system described by the differential equation

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{2} & -a \end{bmatrix} \cdot \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

with $a_{nominal} = 2$. In the first case considered,

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{4} \end{bmatrix}, \quad R = 0.1, \quad \text{and} \quad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} ;$$

in the second case

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad R = 0.1, \quad \text{and} \quad \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} .$$

In both cases $t_o = 0$, $t_f = \infty$ and $F = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$.

First Case. For a cost functional of the type given in equation (6), i.e.,

$$J_1(u) = \frac{1}{2} \int_0^\infty (x_1^2(t) + 0.25x_2^2(t) + 0.1u^2(t))dt ,$$

the optimal control is $u^*(t) = -10k_{12}x_1(t) - 10k_{22}x_2(t)$, where

$$1 - k_{12} - 10k_{12}^2 = 0$$

$$k_{11} - 2k_{12} - 1/2k_{22} - 10k_{12}k_{22} = 0$$

$$1/4 + 2k_{12} - 10_{22}^2 - 4k_{22} = 0 \qquad (8)$$

There is only one solution set of (8) which satisfies the requirement that

$$\begin{bmatrix} k_{11} & k_{12} \\ k_{12} & k_{22} \end{bmatrix}$$

be positive definite. This solution set is $k_{11} = 1.00264693$,

$k_{12} = 0.27015621$, and $k_{22} = 0.14500904$. Thus,

$$u^*(t) = -2.7015621\, x_1(t) - 1.4500904\, x_2(t) \quad .$$

This will be called the "standard" optimal control.

When the cost functional is as given in equation (2), i.e.,

$$J(u) = \frac{1}{2} \int_0^\infty (x_1{}^2(t) + 0.25x_2^2(t) + 0.1u^2(t) + s_1^2(t) + s_2^2(t))dt,$$

the optimal control is $u^*(t) = -10[0\ 1]K_{11}\underline{x}(t) - 10[0\ 1]K_{12}\underline{s}(t)$ where K_{11} and K_{12} satisfy (5) and

$$\underline{\dot{s}}(t) = \begin{bmatrix} \dot{s}_1(t) \\ \dot{s}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{2} & -a \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \quad .$$

Since $K_{11} = K_{11}{}^T$ and $K_{22} = K_{22}{}^T$, let

$$K_{11} = \begin{bmatrix} y_1 & y_2 \\ y_2 & y_3 \end{bmatrix}, \quad K_{12} = \begin{bmatrix} y_4 & y_5 \\ y_6 & y_7 \end{bmatrix} \quad \text{and} \quad K_{22} = \begin{bmatrix} y_8 & y_9 \\ y_9 & y_{10} \end{bmatrix} \quad .$$

Substitution into (5) and reduction yields the following ten scalar equations

$$1 - y_2 - 10y_2^2 = 0$$

$$y_1 - 2y_2 - \frac{1}{2}y_3 - y_5 - 10y_2y_3 = 0$$

$$2y_2 - 4y_3 - 2y_7 + 1/4 - 10y_3^2 = 0$$

$$-1/2y_5 - 1/2y_6 - 10y_2y_6 = 0$$

$$y_4 - 2y_5 - 1/2y_7 - 10y_2y_7 = 0$$

$$y_4 - 2y_6 - 1/2y_7 - y_9 \quad 10y_3y_6 = 0$$

$$y_6 - 2y_7 + y_5 - 2y_7 - y_{10} - 10y_3y_7 = 0$$

$$1 - y_9 - 10y_6^2 = 0$$

$$y_8 - 2y_9 - 1/2y_{10} - 10y_6y_7 = 0$$

$$2y_9 - 4y_{10} + 1 - 10y_7^2 = 0 \ .$$

To solve these equations a simple iterative routine was implemented on a digital computer. This scheme converged to only one solution set. This solution set satisfied the requirement that

$$\begin{bmatrix} K_{11} & K_{12} \\ K_{12}^T & K_{22} \end{bmatrix} \quad \text{be positive definite.}$$

This solution set is:

$y_1 = 1.49747$

$y_2 = 0.270156$

$y_3 = 0.162524$

$y_4 = 0.675191$

$y_5 = 0.436786$

$y_6 = -0.0682145$

$y_7 = -0.061964$

$y_8 = 2.3077$

$y_9 = 0.953467$

$y_{10} = 0.717134.$

Thus,

$$u^*(t) = -2.70156x_1(t) - 1.62524x_2(t) + 0.682145s_1(t) + 0.61964s_2(t) .$$

This will be called the "sensitivity" optimal control.

As before, to check the optimallity of these controls the system was simulated on an analog computer. Control functions of the forms

$$u(t) = -g_1x_1(t) - g_2x_2(t) \quad \text{and} \quad u(t) = -g_1x_1(t)-g_2x_2(t)-g_3s_1(t)-g_4s_2(t)$$

were generated. For the standard optimum case the calculated optimal control results in a local minimum of the cost functional. The calculated optimal control for the sensitivity optimal case does not result in a local minimum of the cost functional. There are values for g_1, g_2, g_3, g_4 which produce a lower value for the cost functional than that which the calculated optimal solution produces.

Second Case. The system for this case is the same as the previous case but the weighting matrix, Q, changes to

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}.$$

The standard optimal control is

$u^*(t) = -10k_{12}x_1(t) - 10k_{22}x_2(t)$ where k_{12} and k_{22} satisfy

$$1 - k_{12} - 10k_{12}^2 = 0$$

$$k_{11} - 2k_{12} - 1/4\ k_{22} - 10k_{12}k_{22} = 0$$

$$2k_{12} - 4k_{22} - 10k_{22}^2 = 0$$

Proceeding as before yields

$$u^*(t) = -2.7015621\ x_1(t) - 1.0664514\ x_2(t)\ .$$

The sensitivity optimum control is

$u^*(t) = -10[0\ 1]K_{11}\underline{x}(t) - 10[0\ 1]K_{12}\underline{s}(t)$ where K_{11} and K_{12} satisfy (5) . Proceeding as in the first case yields,

$$u^*(t) = -2.70156x_1(t) - 1.26865x_2(t) + 0.699306s_1(t) + 0.640484s_2(t)\ .$$

As with the first case, the standard optimum control yielded the minimum of the performance index on the analog computer simulation. However the sensitivity optimum control failed to produce the minimum of the index given in equation (2).

THE ROOT SENSITIVITY

The root sensitivity of the various systems will now be calculated. These will be used as a measure of system sensitivity. The root sensitivity, S_a^r , of system root r with respect to system parameter a is defined to be $S_a^r = \frac{\Delta r}{\Delta a}$ where Δr is the change in root r corresponding to a small change, Δa, in system parameter Δa .

First Order System

Signal flow graphs for the standard optimum and sensitivity optimum systems are shown in Fig. 1a and b respectively. The system characteristic equations for the standard and sensitivity optimum systems are respectively:

$$s + (a + 0.414\ ...) = 0$$

$s^2 + (a + 1.545082)s + (a + 0.738671) = 0$. Clearly, for the standard optimum the root sensitivity is $S_a^r = -1 = 1.0\ \underline{/-180°}$.

For the sensitivity optimum the roots are:

for a = 1.0 $\qquad r_{1,2} = -1.272550 \pm j\ 0.345378$

for a = 1.1 $\qquad r_{1,2} = -1.322549 \pm j\ 0.399218$

Therefore, $S_a^{r_1} = -0.5 - j\ 0.4616 = 0.683\ \underline{/-137°}$.

Thus a significant improvement in the root sensitivity of the system is obtained for the sensitivity optimum system over the standard optimum system.

Second Order Systems

First Case. Signal flow graphs for the standard and sensitivity optimum systems are shown in Fig. 2a and b respectively. The system characteristic equations for the standard and sensitivity optimum systems are respectively:

$$s^2 + (a+1.4500904s + 3.2015621) = 0$$

$$s^4+(a+3.62524)s^3+(2a+7.57148)s^2+(0.5a+7.89789)s + 1.6007 = 0 \quad .$$

The roots for the standard optimum system are:

for $a = 2.0 \quad r_1, \hat{r}_1 = -1.725045 \pm j\,0.475164$

for $a = 2.1 \quad r_1, \hat{r}_1 = -1.775045 \pm j\,0.225337$.

Thus, $S^{r_1}_a = -0.5000 - j\,2.49827 = 2.54781 \,\underline{/-101}^\circ 20'$. The roots for the sensitivity optimum system are:

for $a = 2.0 \quad r_1, \hat{r}_1 = -2.065392 \pm j\,0.904355$

$r_3 = -0.252105$

$r_4 = -1.242309$

for $a = 2.1 \quad r_1, \hat{r}_1 = -2.134190 \pm j\,0.832867$

$r_3 = -0.252200$

$r_4 = -1.204618$

Thus,

$$S^{r_1}_a = -0.68798 - j\,0.71488 = 0.9957 \,\underline{/-133^\circ\ 30'}$$

$$S^{r_3}_a = -0.00095 = 0.00095\,\underline{/-180^\circ}$$

$$S^{r_4}_a = 0.37691 = 0.37691 \,\underline{/0^\circ}$$

Fig. 3 shows root locations for the standard and sensitivity optimum systems. The arrows indicate the direction and relative magnitude of root motion when a increases from a value of 2.0.

For the sensitivity optimum system the real roots correspond to these complex conjugate roots and it is seen that even in this non-optimum case there is an almost three fold reduction in root sensitivity.

Second Case. Signal flow graphs for the standard and sensitivity optimum systems are shown in Fig. 4a and b respectively. The system characteristic equations for the standard and sensitivity optimum system are respectively:

$$s^2 + (a + 1.0664514)s + 3.2015621 = 0$$

$$s^4 + (a+3.26865)s^3+(2a+6.87934)s^2+(0.500a+7.73676)s+1.60078 = 0$$

The roots for the standard optimum system are:

$$\text{for } a = 2.0 \quad r, \hat{r} = -1.533225 \pm j\,0.922378$$

$$\text{for } a = 2.1 \quad r, \hat{r} = -1.583225 \pm j\,0.833642$$

$$\text{Thus,} \quad S_a^r = -0.5000 - j\,0.88736 = 1.018534 \; \underline{/-119°20'} \; .$$

The roots for the sensitivity optimum system are

$$\text{for } a = 2.0 \quad r_1, \hat{r}_1 = -1.829872 \pm j\,1.139461$$

$$r_3 = -0.255039$$

$$r_4 = -1.353815$$

$$\text{for } a = 2.1 \quad r_1, \hat{r}_1 = -1.895791 \pm j\,1.081598$$

$$r_3 = -0.254213$$

$$r_4 = -1.322802$$

$$\text{Thus,} \quad S_a^{r_1} = -0.65919 - j\,0.57863 = 0.97712263 \; \underline{/-138°40'}$$

$$S_a^{r_3} = 0.00826 = 0.00826 \; \underline{/0°}$$

$$S_a^{r_4} = 0.31013 = 0.31013 \; \underline{/0°}$$

Fig. 5 shows root locations for the standard and sensitivity optimum systems. The arrows indicate the direction and relative magnitude of root motion when a increases from a value of 2.0.

As in the first case the complex conjugate roots dominate the sensitivity optimum system. For this case the magnitude of the improvement in root sensitivity performance is less than that obtained in the first case but the sensitivity optimum system does represent an improvement over the standard optimum system.

CONCLUSIONS

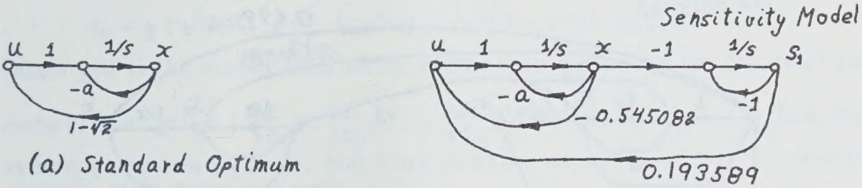
For the first order system the optimal control with respect to the cost functional given in equation (2) which includes the sensitivity variable has been obtained and yields an improvement in root sensitivity over the system using a control that is optimal with respect to the standard cost functional given in equation (6). In the case of the second order system it was found that feeding back the sensitivity variable s reduced the root sensitivity over a system using a control optimal with respect to the cost functional given in equation (2).

REFERENCES

[1] Kahne, Stephen J., "Low Sensitivity Design of Optimal Linear Control Systems," Proceedings of the Fourth Allerton Conference, 1966, pp 499-507.

[2] Athans, Michael and Peter Falb, Optimal Control, McGraw-Hill Book Company, New York, 1966. 879 pp.

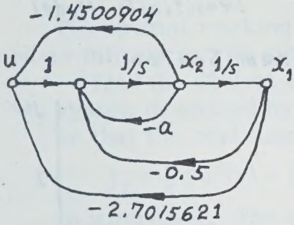
[3] Dompe, Robert J., "A Study of the Design of Optimal Low Sensitivity Linear Control Systems," unpublished Master's thesis, University of Santa Clara, Santa Clara, California, 1967.


Sensitivity Model
u
1
1/s
x
-a
1-√2
u
1
1/s
x
-1
1/s
S_1
-a
-1
-0.545082
0.193589

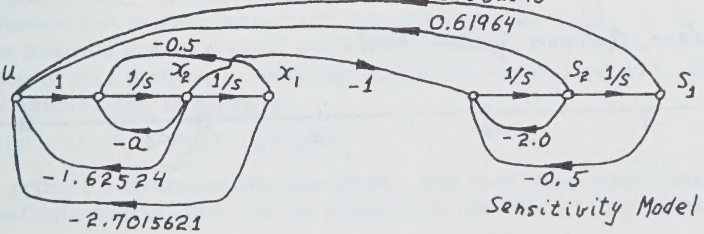
(a) Standard Optimum

(b) Sensitivity Optimum

Fig. 1


-1.4500904
u
1
1/s
x_2
1/s
x_1
-a
-0.5
-2.7015621

(a) Standard Optimum System


0.682145
0.61964
-0.5
u
1
1/s
x_2
1/s
x_1
-1
1/s
S_2
1/s
S_1
-a
-1.62524
-2.7015621
-2.0
-0.5
Sensitivity Model

(b) Sensitivity Optimum System

Fig. 2

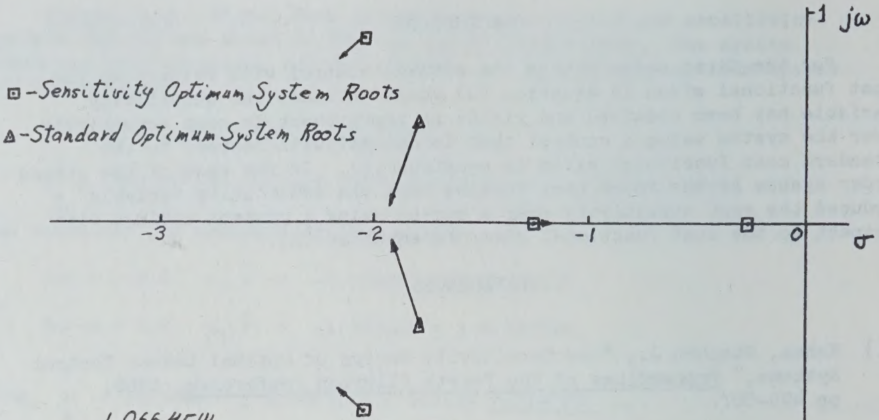
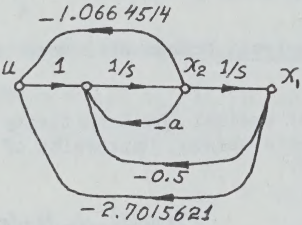
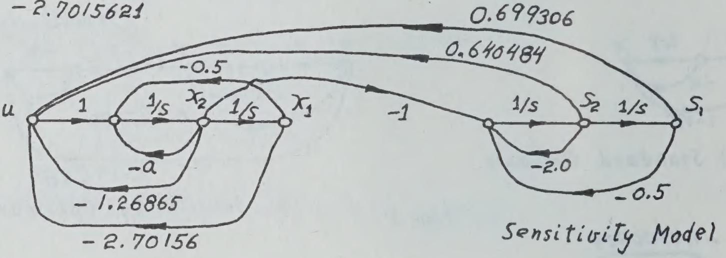


Fig. 3



-1.0664514
u
1
1/s
x2
1/s
x1
-a
-0.5
-2.7015621

(a) Standard Optimum System



0.699306
0.640484
-0.5
u
1
1/s
x2
1/s
x1
-1
1/s
S2
1/s
S1
-a
-2.0
-0.5
-1.26865
-2.70156
Sensitivity Model

(b) Sensitivity Optimum System

Fig. 4

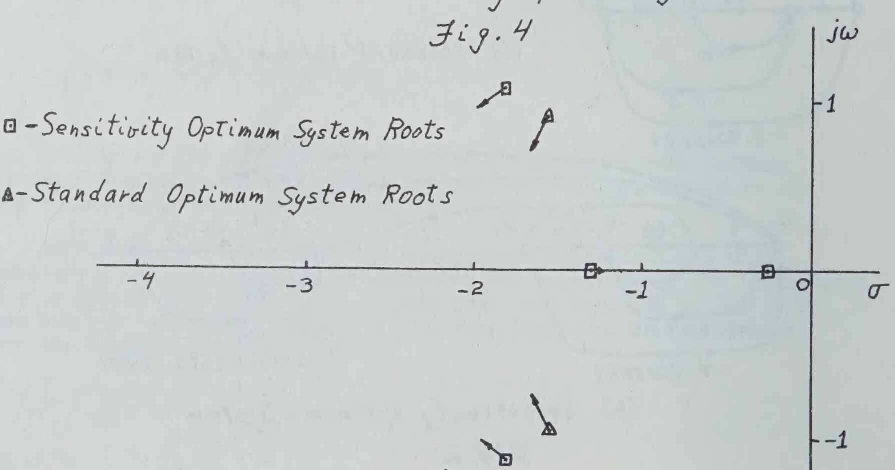


Fig. 5

MINIMUM SENSITIVITY DESIGN OF OPTIMAL LINEAR TRACKING SYSTEMS

M. K. MASTEN and B. F. WOMACK
Department of Electrical Engineering
The University of Texas
Austin, Texas 78712

ABSTRACT

The solution to the optimal tracking problem considers the system described by $\dot{\underline{x}}(t) = A\underline{x}(t) + Bu(t)$, $y(t) = C\underline{x}(t)$, and determines a control signal, $u^*(t)$, which minimizes a cost functional of the form,

$$J_1 = \tfrac{1}{2} Fe^2(T) + \tfrac{1}{2} \int_{t_o}^{T} [Qe^2(t) + Ru^2(t)]dt$$

where $e(t) = y(t) - y_d(t)$ is the difference between the system output and a desired output, $y_d(t)$. The optimal control signal, $u^*(t)$, depends upon the accuracy of the system description. A sensitivity analysis of a designed system could be performed to determine the effects of parameter variations within the plant description, but a more desirable approach would be to consider the variation effects while the system is being designed. This paper considers a single input-single output, multiparameter system with zeros in the transfer function, where a control system is determined which will minimize a cost functional of the form

$$J_2 = \tfrac{1}{2} Fe^2(T) + \tfrac{1}{2} \int_{t_o}^{T} [Qe^2(t) + Ru^2(t) + \underline{v}'(t)\, S\underline{v}(t)]dt$$

where $\underline{v}(t)$ is the output sensitivity vector with respect to the (several) parameters, i.e. $\underline{v}(t) = [\frac{\partial y}{\partial q_1}(t) \frac{\partial y}{\partial q_2}(t) \ldots \frac{\partial y}{\partial q_m}(t)]'$ where $q_1, q_2, \ldots q_m$ are the system parameters. F, Q, and R are positive numbers and S is a symmetric positive semi-definite matrix.

INTRODUCTION

The optimal tracking problem as considered by Kalman [1] can be stated as follows for the single input-single output system:

> Find the control signal, $u^*(t)$, for the linear n th order dynamical system described by $\dot{\underline{x}}(t) = A\underline{x}(t) + Bu(t)$, $y(t) = C\underline{x}(t)$ so that the cost functional (IP)
>
> $$J_1 = \tfrac{1}{2} Fe^2(T) + \tfrac{1}{2} \int_{t_o}^{T} [Qe^2(t) + Ru^2(t)]dt$$
>
> is minimized. The error signal, $e(t) = y(t) - y_d(t)$, is the difference between the system output and a desired output, $y_d(t)$. F, Q, and R are positive constants and the terminal time, T, is fixed.

The solution to this problem is well known [2]; the optimal control, $u^*(t)$, exists, is unique, and is given by

$$u^*(t) = -(1/R)\, B'[\underline{g}(t) - K(t)\underline{x}(t)] \tag{1}$$

where the prime (') indicates transposition. The $n \times n$ real, symmetric, positive definite matrix, K(t), is the solution to the Riccati-type matrix differential equation

$$\dot{K}(t) = -K(t)A - A'K(t) + (1/R)K(t)BB'K(t) - C'QC \tag{2}$$

with the boundary condition $K(T) = C'FC$. The $n \times 1$ vector $\underline{g}(t)$ is the solution of the linear differential equation

$$\dot{\underline{g}}(t) = -[A - (1/R)BB'K(t)]' \, \underline{g}(t) - C'QCy_d(t) \tag{3}$$

with the boundary condition $\underline{g}(T) = C'F\,y_d(T)$.

These results show that the optimal control depends intimately upon the system description as given by A, B, and C. There are numerous articles in current literature which consider the effects of plant variations and/or inaccurate descriptions upon system performance. Such studies can be termed sensitivity analysis; the result shows which parameters need to be measured with the greatest precision since the performance deterioration for given perturbations is obtained. A more desirable approach would be to consider the possible variation effects while in the design stage. A logical method for this approach would be incorporation of a sensitivity measure into the IP. Such an approach has been used for the state regulator problem [3,4] for systems with a single troublesome parameter.

This paper considers the design of a low sensitivity optimal tracking system. All of the system parameters are assumed subject to variation so that these results may be applied to many realistic situations. The treatment is restricted to single input-single output systems; however the transfer function may contain zeros.

PROBLEM STATEMENT

Consider the n th order linear dynamic system described by the differential equation

$$\sum_{i=o}^{n} b_i y^{(i)}(t) = \sum_{j=o}^{n-1} a_j u^{(j)}(t) \quad , \; b_n = 1 \tag{4}$$

or by the transfer function (ignoring initial conditions)

$$\frac{Y(s)}{U(s)} = \frac{a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \ldots + a_o}{s^n + b_{n-1}s^{n-1} + \ldots + b_o} \tag{5}$$

where u(t) and y(t) are the scalar input and output signals. The system is specified by the coefficients $a_{n-1}, a_{n-2}, \ldots a_o, b_{n-1}, b_{n-2}, \ldots b_o$ thus these coefficients are the system parameters, denoted by the m x 1 vector, $\underline{q} = [a_{n-1} a_{n-2} \ldots a_o \; b_{n-1} b_{n-2} \ldots b_o]'$ where $m = 2n$.

The desired output for the plant is given by a prespecified time function $y_d(t)$. The objective is to determine a control signal so that the error, $e(t) = y(t) - y_d(t)$, is small. However the system parameters, $\underline{q}$, are subject to variation. Sensitivity with respect to this variation should be small so that the optimal properties of the system will be maintained. Thus the objective is the determination of a control signal, $u^*(t)$, so that the error is small and also insensitive to plant parameter variations. A measure of the error sensitivity can be the error sensitivity functions, $\partial e/\partial q_i(t)$. However $y_d(t)$ is not dependent upon the system parameters so $\partial e/\partial q_i(t) = \partial y/\partial q_i(t)$. An appropriate IP could be of the form

$$J_2 = \tfrac{1}{2}\, Fe^2(T) + \tfrac{1}{2} \int_{t_o}^{T} [Qe^2(t) + Ru^2(t) + \underline{v}'(t)\, S\underline{v}(t)]dt \tag{6}$$

where the m x 1 vector $\underline{v}(t)$ is the output sensitivity vector with elements $\partial y/\partial q_i(t)$ and S is a m x m symmetric positive semi-definite weighting matrix.

The values of Q, R, and S will determine the relative importance assigned to mismatch error, control effort, and sensitivity, respectively.

SOLUTION

Pontryagin's minimum principle will be used to solve the above problem. The cost functional is specified and the system constraints are defined by the linear differential equation. It is expedient to convert the differential equation into state variable form for incorporation into the IP. Likewise the constraints upon the sensitivity functions must be described and included in the IP.

Assignment of State Variables and Sensitivity Constraints

There are numerous techniques for assigning state variables for the system described in equation (4). A possible technique is to define

$$\begin{aligned}
x_1(t) &= y(t)\\
x_2(t) &= \dot{y}(t) - h_1 u(t)\\
x_3(t) &= \ddot{y}(t) - h_1 \dot{u}(t) - h_2 u(t)\\
&\vdots\\
x_n(t) &= y^{(n-1)}(t) - h_1 u^{(n-2)}(t) - h_2 u^{(n-3)}(t) - \ldots - h_{n-1} u(t)
\end{aligned} \tag{7}$$

where $h_1 = a_{n-1}$, $h_2 = a_{n-2} - h_1 b_{n-1}$, $h_3 = a_{n-3} - h_1 b_{n-2} - h_2 b_{n-1}$, ... Then

$$\begin{aligned}
\dot{x}_1(t) &= x_2(t) + h_1 u(t)\\
\dot{x}_2(t) &= x_3(t) + h_2 u(t)\\
&\vdots\\
\dot{x}_n(t) &= -\sum_{i=o}^{n-1} b_i x_{i+1}(t) + h_n u(t)
\end{aligned} \tag{8}$$

where $h_n = a_o - \sum_{i=1}^{n-1} b_i h_i$. This system of equations can be written as

$$\begin{bmatrix} \dot{x}_1(t)\\ \dot{x}_2(t)\\ \vdots\\ \dot{x}_n(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0\\ 0 & 0 & 1 & \ldots & 0\\ & & \vdots & & \\ -b_o & -b_1 & -b_2 & \ldots & -b_{n-1} \end{bmatrix} \begin{bmatrix} x_1(t)\\ x_2(t)\\ \vdots\\ x_n(t) \end{bmatrix} + \begin{bmatrix} h_1\\ h_2\\ \vdots\\ h_n \end{bmatrix} u(t) \tag{9}$$

or more briefly as

$$\begin{aligned}
\underline{\dot{x}}(t) &= A\underline{x}(t) + Bu(t)\\
y(t) &= C\underline{x}(t)
\end{aligned} \tag{10}$$

where C = [1 0 0 0 0 .... 0]. This scheme has the advantage that A and B are functions of the system parameters whereas C is independent of $\underline{q}$. Therefore the output sensitivity vector $\underline{v}(t)$ can be written

$$\underline{v}(t) = \begin{bmatrix} \partial y/\partial q_1\\ \partial y/\partial q_2\\ \vdots\\ \partial y/\partial q_m \end{bmatrix} = \begin{bmatrix} c_1 \partial x_1/\partial q_1 + c_2 \partial x_2/\partial q_1 + \ldots c_n \partial x_n/\partial q_1\\ c_1 \partial x_1/\partial q_2 + c_2 \partial x_2/\partial q_2 + \ldots c_n \partial x_n/\partial q_2\\ \vdots\\ c_1 \partial x_1/\partial q_m + c_2 \partial x_2/\partial q_m + \ldots c_n \partial x_n/\partial q_m \end{bmatrix} \tag{11}$$

Define the n×m state variable sensitivity matrix, $\underline{w}(t)$, by

$$\underline{w}(t) = \nabla_{\underline{q}}\underline{x} = \begin{bmatrix} \partial x_1/\partial q_1 & \partial x_1/\partial q_2 & \cdots & \partial x_1/\partial q_m \\ \partial x_2/\partial q_1 & \partial x_2/\partial q_2 & \cdots & \partial x_2/\partial q_m \\ \vdots & & & \\ \partial x_n/\partial q_1 & \partial x_n/\partial q_2 & \cdots & \partial x_n/\partial q_m \end{bmatrix} \tag{12}$$

where $\nabla_{\underline{q}}\underline{x}$ is the gradient of the vector $\underline{x}$ with respect to the vector $\underline{q}$. Then the output sensitivity vector can be written as

$$\underline{v}(t) = \underline{w}'(t)C' \tag{13}$$

Thus the determination of the state variable sensitivity functions will provide the output sensitivity functions.

To determine the state variable sensitivity functions, consider the single input system described by the $n \times 1$ vector state equation (for a given time variable input)

$$\dot{\underline{x}} = \underline{f}(\underline{x}, \underline{q}, t), \quad \underline{x}(t_o) = \underline{x}_o \tag{14}$$

Rearranging and differentiating,

$$d\dot{\underline{x}} - d\underline{f}(\underline{x}, \underline{q}, t) = \underline{0} \tag{15}$$

or

$$d\dot{\underline{x}} - (\nabla_{\underline{x}}\underline{f})\, d\underline{x} - (\nabla_{\underline{q}}\underline{f})\, d\underline{q} - \partial\underline{f}/\partial t\, dt = \underline{0} \tag{16}$$

The solution to the state equation can be written as

$$\underline{x} = \underline{x}(t, \underline{q}, \underline{x}_o) \tag{17}$$

which shows the functional dependence upon $t, \underline{q}$, and $\underline{x}_o$ (where $\underline{x}_o$ is the value of $\underline{x}$ at $t = t_o$). Thus

$$d\underline{x} = (\partial\underline{x}/\partial t)\, dt + (\nabla_{\underline{q}}\underline{x})\, d\underline{q} + (\nabla_{\underline{x}_o}\underline{x})\, d\underline{x}_o \tag{18}$$

Likewise

$$d\dot{\underline{x}} = (\partial\dot{\underline{x}}/\partial t)\, dt + (\nabla_{\underline{q}}\dot{\underline{x}})\, d\underline{q} + (\nabla_{\underline{x}_o}\dot{\underline{x}})\, d\underline{x}_o \tag{19}$$

Substituting these expressions for $d\underline{x}$ and $d\dot{\underline{x}}$ into (16) and collecting terms,

$$[\partial\dot{\underline{x}}/\partial t - (\nabla_{\underline{x}}\underline{f})\, \partial\underline{x}/\partial t - \partial\underline{f}/\partial t]\, dt + [\nabla_{\underline{q}}\dot{\underline{x}} - (\nabla_{\underline{x}}\underline{f})\, \nabla_{\underline{q}}\underline{x} - \nabla_{\underline{q}}\underline{f}]\, d\underline{q}$$

$$+ [\nabla_{\underline{x}_o}\dot{\underline{x}} - (\nabla_{\underline{x}}\underline{f})\, \nabla_{\underline{x}_o}\underline{x}]\, d\underline{x}_o = \underline{0} \tag{20}$$

If $\underline{q}$, t, and $\underline{x}_o$ are independent variables (i.e. slow variation), the coefficient expressions for $d\underline{q}$, dt, and $d\underline{x}_o$ must each be zero. Also $\nabla_{\underline{q}}\dot{\underline{x}} = d/dt\, (\nabla_{\underline{q}}\underline{x})$ $\underline{\underline{\Delta}}$ $\dot{\underline{w}}(t)$ if $\underline{q}$ and t are independent. Thus the coefficient for $d\underline{q}$, set to zero, can be written as

$$\dot{\underline{w}} = (\nabla_{\underline{x}}\underline{f})\, \underline{w} + \nabla_{\underline{q}}\underline{f} \tag{21}$$

This is the sensitivity equation for determining the state variable sensitivity functions with respect to the system parameters $\underline{q}$. If the initial conditions for the case of $\underline{q} = \underline{q}_o$ and $\underline{q} = \underline{q}_o + \Delta\underline{q}$ are identical, then for $t = t_o$ there is no deviation in the two solutions. Therefore all initial conditions for solving this sensitivity equation are zero, i.e. $\underline{w}(t_o) = \underline{0}$.

For the linear system of (10), $\underline{f}(\underline{x},\underline{q},t) = A\underline{x}(t) + Bu(t)$. So $\nabla_{\underline{x}}\underline{f} = A$. The elements of $\nabla_{\underline{q}}\underline{f}$ can be written as

$$\partial f_i/\partial q_j = (\partial a_{i1}/\partial q_j)x_1 + (\partial a_{i2}/\partial q_j)x_2 + \ldots(\partial a_{in}/\partial q_j)x_n + (\partial b_i/\partial q_j)u \tag{22}$$

Partitioning A into n column submatrices,

$$A = [A_1 \mid A_2 \mid A_3 \mid \ldots \mid A_n] \tag{23}$$

where $A_i = [a_{1i}a_{2i}\ldots a_{ni}]'$ enables $\nabla_{\underline{q}}\underline{f}$ to be written as

$$\nabla_{\underline{q}}\underline{f} = x_1\nabla_{\underline{q}}A_1 + x_2\nabla_{\underline{q}}A_2 + \cdots + x_n\nabla_{\underline{q}}A_n + u\nabla_{\underline{q}}B \tag{24}$$

where $x_1, x_2, \ldots x_n$ and u are scalar functions. Therefore the sensitivity equation for the linear system of (10) can be written as

$$\underline{\dot{w}}(t) = A\underline{w}(t) + \sum_{i=1}^{n}(\nabla_{\underline{q}}A_i)x_i(t) + (\nabla_{\underline{q}}B)u(t), \quad \underline{w}(t_o) = \underline{0} \tag{25}$$

Note that the system matrix, A, is identical for both $\underline{x}(t)$ and $\underline{w}(t)$. The terms $(\nabla_{\underline{q}}B)u(t)$ and $\sum_{i=1}^{n}(\nabla_{\underline{q}}A_i)x_i(t)$ represent forcing functions. If the system under consideration has no zeros in its transfer function, then the vector B is independent of the system parameters and $(\nabla_{\underline{q}}B)$ vanishes. The resulting sensitivity equation has often been used in sensitivity studies and is the basis of the "sensitivity points" [5] method for determining sensitivity functions.

The system state equations given in (10) are in vector form. The use of a costate vector will enable this constraint to be incorporated into the scalar IP. However equation (25) is a matrix constraint and therefore not easily incorporated into a scalar IP. In this particular problem, the importance is placed on the output sensitivity functions given by $\underline{w}'(t)C'$. Only the first element of C is non-zero therefore only the sensitivity functions for the first state variable are of interest; the remaining elements of $\underline{w}(t)$ are unnecessary information. This realization enables (25) to be reduced to a vector equation for $\underline{v}(t)$. This conversion is best illustrated by an example. Consider the system with the transfer function

$$T(s) = \frac{as + b}{s^2 + cs + d} \tag{26}$$

Assigning the state variables as $x_1 = y$, $x_2 = \dot{y} - au$, then

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -d & -c \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} a \\ (b-ca) \end{bmatrix} u(t) \tag{27}$$

The state variable sensitivity equation is

$$\underline{\dot{w}}(t) = A\underline{w}(t) + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -x_2(t) & -x_1(t) \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 & 0 \\ -c & 1 & -a & 0 \end{bmatrix} u(t) \tag{28}$$

where

$$\underline{w}(t) = \begin{bmatrix} w_{1a}(t) & w_{1b}(t) & w_{1c}(t) & w_{1d}(t) \\ w_{2a}(t) & w_{2b}(t) & w_{2c}(t) & w_{2d}(t) \end{bmatrix} \tag{29}$$

with $w_{1a}(t) = \partial x_1/\partial a(t)$, $w_{2a}(t) = \partial x_2/\partial a(t)$, etc. The simulation for the two equations of (27) and the eight equations of (28), after some rearrangement, is given in Figure 1. Observation of the figure shows that the equations for $\underline{v}(t) = [v_a(t)\ v_b(t)\ v_c(t)\ v_d(t)]'$ can be written as

$$\dot{\underline{v}}(t) = E\underline{v}(t) + fu(t) + G\underline{x}(t) \tag{30}$$

where

$$E = \begin{bmatrix} -c & -d & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -c & -d \\ 0 & 0 & 1 & 0 \end{bmatrix},\ f = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix},\ G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & 0 \end{bmatrix} \tag{31}$$

This is now a vector constraint for the sensitivity functions and can be incorporated into the IP by use of a sensitivity costate vector.

Minimization of Cost Functional

Substitution of $e(t) = C\underline{x}(t) - y_d(t)$ makes the IP an explicit function of the states and the sensitivity functions; the optimal control will be a function of $\underline{x}(t)$ and $\underline{v}(t)$; therefore observability will be a system constraint to be placed on (10) and (30). Utilization of a costate vector $\underline{p}(t)$ and a sensitivity costate vector $\underline{r}(t)$ results in the following Hamiltonian:

$$\begin{aligned} H = \tfrac{1}{2}Q[C\underline{x}(t) - y_d(t)]^2 + \tfrac{1}{2}Ru^2(t) + \tfrac{1}{2}\underline{v}'(t)S\underline{v}(t) + \underline{p}'(t)[A\underline{x}(t) + Bu(t)] \\ + \underline{r}'(t)[E\underline{v}(t) + fu(t) + G\underline{x}(t)] \end{aligned} \tag{32}$$

Using the minimum principle, the optimal control is given by

$$\partial H/\partial u = 0 = Ru(t) + B'\underline{p}(t) + f'\underline{r}(t) \tag{33}$$

or

$$u^*(t) = -(1/R)\,[B'\underline{p}(t) + f'\underline{r}(t)] \tag{34}$$

and the canonical equations yield;

$$\begin{aligned} \dot{\underline{p}}(t) &= -\partial H/\partial\underline{x}(t) = -C'Q\,C\underline{x}(t) + C'Qy_d(t) - A'\underline{p}(t) - G'\underline{r}(t) \\ \dot{\underline{x}}(t) &= \partial H/\partial\underline{p}(t) = A\underline{x}(t) + Bu(t) = A\underline{x}(t) - (1/R)[BB'\underline{p}(t) + Bf'\underline{r}(t)] \\ \dot{\underline{r}}(t) &= -\partial H/\partial\underline{v}(t) = -S\underline{v}(t) - E'\underline{r}(t) \\ \dot{\underline{v}}(t) &= \partial H/\partial\underline{r}(t) = E\underline{v}(t) + fu(t) + G\underline{x}(t) \\ &= E\underline{v}(t) + G\underline{x}(t) - (1/R)[fB'\underline{p}(t) + ff'\underline{r}(t)] \end{aligned} \tag{35}$$

with the boundary conditions,

$$\begin{aligned} \underline{x}(t_o) &= \underline{x}_o \\ \underline{v}(t_o) &= \underline{0} \\ \underline{p}(T) &= \partial/\partial\underline{x}\,[\tfrac{1}{2}\,Fe^2(T)] = C'F\,C\underline{x}(T) - C'Fy_d(T) \\ \underline{r}(T) &= \underline{0} \end{aligned} \tag{36}$$

The canonical equations can be written in terms of an augmented state vector,

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \hline \dot{\underline{v}}(t) \\ \hline \dot{\underline{p}}(t) \\ \hline \dot{\underline{r}}(t) \end{bmatrix} = \left[\begin{array}{c|c|c|c} A & \underline{0} & -(1/R)BB' & -(1/R)Bf' \\ \hline G & E & -(1/R)fB' & -(1/R)ff' \\ \hline -C'QC & \underline{0} & -A' & -G' \\ \hline \underline{0} & -S & \underline{0} & -E' \end{array}\right] \begin{bmatrix} \underline{x}(t) \\ \hline \underline{v}(t) \\ \hline \underline{p}(t) \\ \hline \underline{r}(t) \end{bmatrix} + \begin{bmatrix} \underline{0} \\ \hline \underline{0} \\ \hline C'Qy_d(t) \\ \hline \underline{0} \end{bmatrix} \tag{37}$$

This is a system of $2(n+m)$ linear differential equations. The above augmented state vector can be partitioned into two $n+m$ components,

$$\underline{\alpha}(t) \triangleq \begin{bmatrix} \underline{x}(t) \\ \hline \underline{v}(t) \end{bmatrix} \qquad \underline{\beta}(t) \triangleq \begin{bmatrix} \underline{p}(t) \\ \hline \underline{r}(t) \end{bmatrix} \tag{38}$$

so that equation (37) becomes

$$\begin{bmatrix} \dot{\underline{\alpha}}(t) \\ \hline \dot{\underline{\beta}}(t) \end{bmatrix} = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & -A_{11}' \end{array}\right] \begin{bmatrix} \underline{\alpha}(t) \\ \hline \underline{\beta}(t) \end{bmatrix} + \begin{bmatrix} \underline{0} \\ \hline D \end{bmatrix} y_d(t) \tag{39}$$

where A_{11}, A_{12}, A_{21}, and D are the proper partitions of the matrices of (37). The boundary conditions of (39) will be

$$\underline{\alpha}(t_o) = \begin{bmatrix} \underline{x}_o \\ \hline \underline{0} \end{bmatrix}; \quad \underline{\beta}(T) = \begin{bmatrix} C'FCx(T) - C'Fy_d(T) \\ \hline \underline{0} \end{bmatrix} = \left[\begin{array}{c|c} C'FC & \underline{0} \\ \hline \underline{0} & \underline{0} \end{array}\right] \begin{bmatrix} \underline{x}(T) \\ \hline \underline{v}(T) \end{bmatrix}$$
$$+ \begin{bmatrix} C'F \\ \hline \underline{0} \end{bmatrix} y_d(T) \triangleq M\underline{\alpha}(T) + Ny_d(T) \tag{40}$$

Let $\varphi(t, t_o)$ be the transition matrix for (39). Then the solution at the final time can be written

$$\begin{bmatrix} \underline{\alpha}(T) \\ \hline \underline{\beta}(T) \end{bmatrix} = \varphi(T, t) \begin{bmatrix} \underline{\alpha}(t) \\ \hline \underline{\beta}(t) \end{bmatrix} + \int_t^T \varphi(T, \tau) \begin{bmatrix} 0 \\ \hline D \end{bmatrix} y_d(\tau)d\tau \tag{41}$$

Partitioning $\varphi(T, t)$ into four $(n+m)$ submatrices

$$\varphi(T, t) \triangleq \left[\begin{array}{c|c} \varphi_{11}(T, t) & \varphi_{12}(T, t) \\ \hline \varphi_{21}(T, t) & \varphi_{22}(T, t) \end{array}\right] \tag{42}$$

and substituting the boundary conditions of (40) one can solve for $\underline{\beta}(t)$.

$$\underline{\beta}(t) = [M\varphi_{12}(T, t) - \varphi_{22}(T, t)]^{-1} \cdot [\{\varphi_{21}(T, t) - M\varphi_{11}(T, t)\}\underline{\alpha}(t) - Ny_d(T)$$
$$+ \int_t^T \{\varphi_{22}(T, \tau) - M\varphi_{12}(T, \tau)\} Dy_d(\tau)\, d\tau] \tag{43}$$

which suggests the relation

$$\underline{\beta}(t) = K(t)\underline{\alpha}(t) - \underline{g}(t) \tag{44}$$

if the inverse exists; the proof of this existence is given by Kalman [6]. The significance of (44) is recognized when the expression for the optimal control given in (34) is rewritten as

$$u^*(t) = -(1/R)[B' \mid f'] \begin{bmatrix} \underline{p}(t) \\ \hline \underline{r}(t) \end{bmatrix} = -(1/R)[B' \mid f']\underline{\beta}(t) \tag{45}$$

and using the results of (44)

$$u^*(t) = -(1/R)\{[B'K_{11}(t) + f'K_{21}(t)]\underline{x}(t) + [B'K_{12}(t) + f'K_{22}(t)]\underline{v}(t) - [B' \;\vdots\; f']\,\underline{g}(t)\} \tag{46}$$

Thus the optimal control utilizes state variable and sensitivity function feedback as well as the signal $\underline{g}(t)$.

Optimal Control Characteristics

Substituting equation (44) into equation (39) yields

$$\dot{\underline{\alpha}}(t) = [A_{11} + A_{12}\,K(t)]\underline{\alpha}(t) - A_{12}\underline{g}(t) \tag{47}$$

and

$$\dot{\underline{\beta}}(t) = [A_{21} - A_{11}'\,K(t)]\underline{\alpha}(t) + A_{11}'\,\underline{g}(t) + Dy_d(t) \tag{48}$$

Differentiating (44) with respect to time and substituting the expression for $\dot{\underline{\alpha}}(t)$ from (47) yields

$$\dot{\underline{\beta}}(t) = [\dot{K}(t) + K(t)A_{11} + K(t)\,A_{12}K(t)]\underline{\alpha}(t) - \dot{\underline{g}}(t) - K(t)\,A_{12}\,\underline{g}(t) \tag{49}$$

Comparing this expression with equation (48) shows

$$\dot{K}(t) = -K(t)\,A_{11} - A_{11}'\,K(t) - K(t)\,A_{12}\,K(t) + A_{21} \tag{50}$$

and

$$\dot{\underline{g}}(t) = (-A_{11}' - K(t)\,A_{12})\,\underline{g}(t) - D\,y_d(t) \tag{51}$$

Note that $\varphi_{11}(T,T) = \varphi_{22}(T,T) = I$ and $\varphi_{12}(T,T) = \varphi_{21}(T,T) = \underline{0}$. Thus (43) implies that $\underline{\beta}(T) = M\,\underline{\alpha}(T) + Ny_d(T)$ and (44) implies $\underline{\beta}(T) = K(T)\,\underline{\alpha}(T) - \underline{g}(T)$. These must be valid for arbitrary $\underline{\alpha}(T)$ and $y_d(T)$ thus the boundary conditions for (50) and (51) are

$$K(T) = M \quad \text{and} \quad \underline{g}(T) = -\,Ny_d(T) \tag{52}$$

The differential equation for $K(t)$ is recognized to be of the Riccati type. Note that the value of $K(t)$ will be independent of $y_d(t)$, thus the state variable and sensitivity function feedback gains are completely specified by the elements of the IP and the system description. Recall also that the Riccati solution is symmetric, i.e. $K(t) = K'(t)$ thus $K_{12}(t) = K_{21}(t)$ thus simplifying the determination of the optimal control given in (46).

Observation of (39) reveals $A_{12} = A_{12}'$ so (51) can be rewritten as

$$\dot{\underline{g}}(t) = -[A_{11} + A_{12}\,K(t)]'\,\underline{g}(t) - D\,y_d(t) \tag{53}$$

Comparison of this equation with (47) shows that the system which generates $\underline{g}(t)$ is the adjoint of the optimal system described by (47). Let $\Psi(t,t_o)$ be the transition matrix for the adjoint system, then the final value, $\underline{g}(T)$, can be written as

$$\underline{g}(T) = \Psi(T,t)\,\underline{g}(t) - \int_t^T \Psi(T,\tau)Dy_d(\tau)d\tau \tag{54}$$

and rearranging

$$\underline{g}(t) = \Psi^{-1}(T,t)[\underline{g}(T) + \int_t^T \Psi(T,\tau)Dy_d(\tau)d\tau] \tag{55}$$

Using the boundary condition given in (52) yields

$$\underline{g}(t) = \Psi^{-1}(T,t)[-Ny_d(T) + \int_t^T \Psi(T,\tau)Dy_d(\tau)d\tau] \tag{56}$$

which shows that the future value of the desired output is required in order that $\underline{g}(t)$ be computed at any given time. That is, the present value of the optimal control depends upon the future values of the desired output. Assuming this condition is fulfilled the initial condition on $\underline{g}(t_o)$ can be seen to be

$$\underline{g}(t_o) = \Psi^{-1}(T,t_o)[-Ny_d(T) + \int_{t_o}^{T} \Psi(T,\tau)Dy_d(\tau)d\tau] \tag{57}$$

Using this value as the initial condition then $\underline{g}(t)$ can be generated forward in time by the system of (53).

Figure 2 incorporates equations (47), (53), and (57) into the overall optimal insensitive tracking system.

CONCLUSIONS

The structure for the optimum insensitive tracker is derived. The design is sufficiently general to enable consideration of possible variations in all the system parameters, and the simulation necessary to generate the appropriate sensitivity functions is derived. The notation used in this paper is that normally associated with time invariant systems; however the conclusions can be extended to time variable situations without loss of generality by replacing A by A(t), B by B(t), etc.

The "classical" optimal tracking system uses state variable feedback and an additional input as determined by the solution to an auxilary differential equation. Insensitive optimal tracking utilizes both state variable and sensitivity function feedback as well as the additional signal, $\underline{g}(t)$. In both cases the desired output must be completely known a priori. The feedback gains for $\underline{x}(t)$ and $\underline{v}(t)$ are determined from partitions of a $(n+m)\times(n+m)$ Riccati equation. In concept, this offers no more difficulty than solution to the conventional matrix Riccati equation. The relative sizes of Q and S will best be determined by a given circumstance. The sensitivity with respect to some parameters may be known a priori to be relatively unimportant. A small value for the appropriate elements of S can then be assigned. The overall objective is to maintain small tracking error; if it is known that parameter variations will be small a small value for S would be best. On the other hand, if the values of A and B are only known approximately, S should be larger.

BIBLIOGRAPHY

1. Kalman, R. E., The Theory of Optimal Control and the Calculus of Variation, Mathematical Optimization Techniques, University of California Press, Berkeley, California, 1963.
2. Athans, M. and Falb, P. L., Optimal Control, McGraw-Hill, N.Y., 1966.
3. Kahne, S. T., Low Sensitivity Design of Optimal Linear Control Systems, Proceedings of 4th Annual Allerton Conference on Circuit and System Theory, October 1966, pp. 499-507.
4. D'Angelo, H. et.al., Trajectory Sensitivity of an Optimal Control System, Proceedings of 4th Annual Allerton Conference on Circuit and System Theory, October 1966, pp. 489-498.
5. Tomovic, R., Sensitivity Analysis of Dynamic Systems, McGraw-Hill, New York, 1963.
6. Kalman, R. E., Contributions to the Theory of Optimal Control, Bol. Soc. Math. Mexico, 1960, pp. 102-119.

ACKNOWLEDGEMENT

Supported by DOD' s JSEP under Grant AF-AFOSR-766-67.

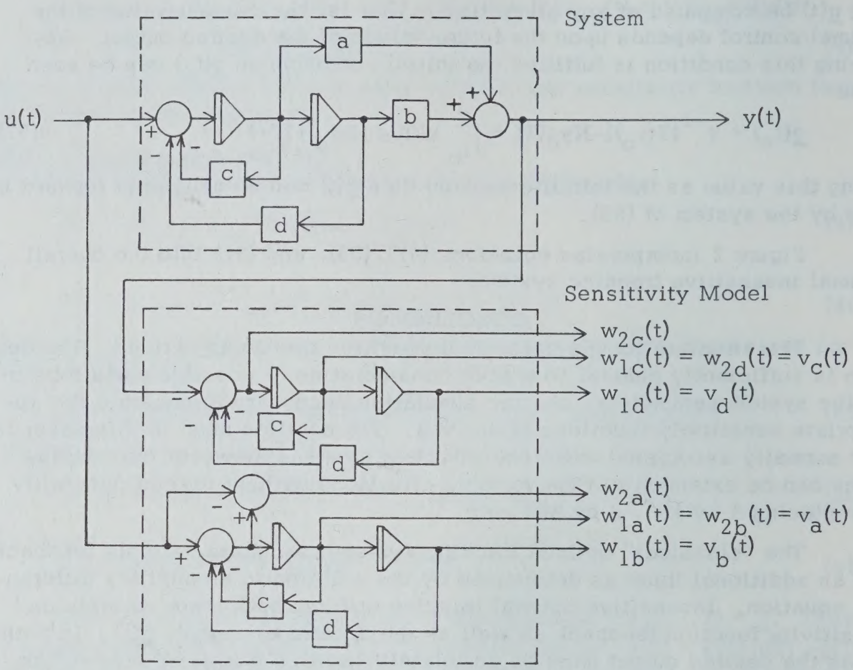
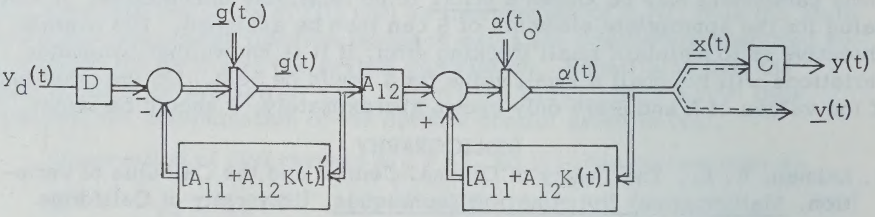

System
u(t)
a
b
c
d
+
−
+
+
y(t)
Sensitivity Model
$w_{2c}(t)$
$w_{1c}(t) = w_{2d}(t) = v_c(t)$
$w_{1d}(t) = v_d(t)$
$w_{2a}(t)$
$w_{1a}(t) = w_{2b}(t) = v_a(t)$
$w_{1b}(t) = v_b(t)$
c
d
c
d
+
−

FIGURE 1


$y_d(t)$
D
$\underline{g}(t_o)$
$\underline{g}(t)$
A_{12}
$\underline{\alpha}(t_o)$
$\underline{\alpha}(t)$
$\underline{x}(t)$
C
y(t)
$\underline{v}(t)$
$[A_{11}+A_{12}K(t)]'$
$[A_{11}+A_{12}K(t)]$
−
+

$$A_{11} = \left[\begin{array}{c|c} A & \underline{0} \\ \hline G & E \end{array}\right]$$

$$A_{12} = -\frac{1}{R}\left[\begin{array}{c|c} BB' & Bf' \\ \hline fB' & ff' \end{array}\right]$$

$$D = \left[\begin{array}{c} C'\,Q \\ \hline \underline{0} \end{array}\right]$$

$$\underline{\alpha}(t) = \left[\begin{array}{c} \underline{x}(t) \\ \hline \underline{v}(t) \end{array}\right] = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \\ v_1(t) \\ v_2(t) \\ \vdots \\ v_m(t) \end{bmatrix}$$

FIGURE 2

ON IDENTIFICATION OF CONSTRAINED DYNAMIC SYSTEMS WITH HIGH DIMENSIONS

MASANAO AOKI [†]
Department of Engineering, University of California, Los Angeles, California

ABSTRACT

This paper considers computationally efficient algorithms for identifying unknown constant transition matrices of discrete-time dynamic systems, where the matrices are assumed a priori to satisfy a set of linear constraint equations.

1. INTRODUCTION

Consider an identification problem of the system S_1 described by

$$x(i+1) = Ax(i) + f(i) \tag{1.1}$$

where x is an n-vector, and where f(i) represents a known forcing term.

Frequently one possesses some a priori knowledge on the system to be identified. A well-known way to incorporate a priori information about the systems is to assume that the system equations have known functional forms parameterized by an unknown parameter θ in a known parameter space and employ either Bayes or the minmax method or proceed by considering an augmented state vector $(x, \theta)^T$ and estimate the augmented state vector in some manner.[1-7] It is assumed in this paper that some of the elements of the matrix A are unknown constants. Thus, the Bayesian approach is excluded from our consideration.[1] Some of the existing techniques of system identification do not take full advantage of the a priori information on A (beyond the mere quantitative requirements such as controllability and observability) and treat all elements of A (or n unknown elements of the phase canonical form of A) equally as if they are all unknown.[2,3] These methods do not take advantage of the fact that some elements may be known. Since the amounts of computation in system identification grow as a nonlinear function of the dimension n, it is practically important to be able to reduce the amount of computation by taking advantage of the information on A which is available a priori, for large n.

This paper presents a way of incorporating the a priori information on the systems to be identified and develops a new system identification algorithm. Since A is usually generated through a set of coupled equations written for individual subsystems comprising the total system and since the process of transforming A into the phase canonical form may result in having matrix elements which are in more complex expressions than the original system parameters to be identified, it is assumed that A is not necessarily in the phase canonical form. It is assumed in this paper either that the matrix A in Equation (1.1) comes from a class of (nxn) matrices satisfying the constraint

$$AQ = 0_{n,k} \tag{1.2a}$$

[†] Supported in part by Research Contract Nonr 233(52) of the Office of Naval Research and by NIH Grant 7M-14633-01.

or (1.2a) and

$$PA = 0_{k,n} \tag{1.2b}$$

where P is a known kxn matrix and where Q is a known nxk matrix, or that A can be written as A = (matrix with known elements only) + (matrix with unknown elements only) where the matrix with unknown elements satisfies constraints of the type indicated by Equations (1.2a) or (1.2a) and (1.2b).† Matrices P and Q are both taken to be of maximal rank. The symbol $0_{i,k}$ is used to indicate the (ixk) null matrix. To present the basic ideas clearly, assume that A is stable.

The identification algorithm to be developed in the main body of the paper makes use of the fact that any matrix A of this class has a representation such as

$$A = A_k A_u \tag{1.3a}$$

or

$$A = A_u A_k \tag{1.3b}$$

when the constraint is given by (1.2a) or (1.2b) respectively, where A_k and A_u are known and unknown respectively, or as

$$A = DEC \tag{1.3c}$$

when the constraint is given by (1.2a) and (1.2b).

The constraints on A of the type (1.2a) or (1.2b) or both are significant in the sense that the unknown matrix A is assumed to be a member of a class of matrices with a certain structure or approximated by a member of such a class. Thus for A with certain known structural properties, identification operations can be performed with no loss of information with a saving of the computational amounts. For other A's the operations involve certain loss of information which may be tolerable in view of the lesser computational requirements. The deterministic problem is considered first where the noise is absent in the descriptions of S_1, then recursive identification procedure with noisy measurements are indicated for some subclasses of A.

2. IDENTIFICATION OF DETERMINISTIC SYSTEMS WITH CONSTRAINT, I

Consider the case where the matrix A satisfies a constraint equation

$$AQ = 0_{n,k} \tag{2.1}$$‡

where Q is a known (nxk) matrix of maximal rank, $k < n$. As shown in Appendix A, Equation (2.1) implies that A is at most of rank ℓ, where $\ell = n - k$, and that A has the factorization

$$A = A_u A_k \tag{2.2}$$

---

† Matrices with known elements may be incorporated in f_i when x_i are all directly observed.

‡ In other words, row vectors of A are orthogonal to the subspace spanned by the column vectors of Q.

where A_u and A_k are $(n\times\ell)$ and $(\ell\times n)$ matrices respectively, and where A_u is unknown and A_k is a known matrix of rank ℓ. Thus the identification of A reduces to that of identifying the $(n\times\ell)$ matrix A_u. Consider now the problem identifying A_u in (2.2).

Exact Measurements

Take the error of identification to be given by

$$J_N = \sum_{i=0}^{N-1} \left\| x_{i+1} - Ax_i - f_i \right\|^2$$

$$= \mathrm{tr}\left\{-AW^T_{N-1,0} - W_{N-1,0}A^T + AZ_{N-1,0}A^T\right\} + \text{const.}$$

where T denotes a transpose and where

$$W_{0,N-1} \triangleq \sum_{i=0}^{N-1} (x_{i+1} - f_i)x_i^T : \text{nxn matrix}$$

$$Z_{0,N-1} \triangleq \sum_{i=0}^{N-1} x_i x_i^T \qquad : \text{nxn matrix} \tag{2.3}$$

and where A is given by (2.2). J_N is to be minimized with respect to A_u. The result of this minimization gives

$$\hat{A}_u(N)\, A_k\, Z_{0,N-1}\, A_k^T = W_{0,N-1}\, A_k^T$$

where $\hat{A}_u(N)$ is the best estimate of A_u based on data $x_0,\dots x_{N-1}$. If N is taken large enough so that $Z_{0,N-1}$ can be assumed to be positive definite, then

$$\hat{A}_u(N) = W_{0,N-1}\, A_k^T (A_k Z_{0,N-1} A_k^T)^{-1} \tag{2.4}$$

and

$$\hat{A}(N) = \hat{A}_u(N)\, A_k$$

From (2.3), it is seen that

$$W_{0,N-1} = A_u\, A_k\, Z_{0,N-1}$$

Thus

$$\hat{A}_u(N) = A_u A_k Z_{0,N-1} A_k^T (A_k Z_{0,N-1} A_k^T)^{-1} = A_u$$

showing that for deterministic systems, A can be identified exactly, provided $Z_N > 0$. Note that in (2.4), an inversion of $(\ell\times\ell)$ matrix is required rather than the inversion of (nxn) matrix which would be required if A is identified directly.[3] Note also the fact that J_N defined by

$$J_N = \sum_{i=0}^{N-1} \left\| x_{i+1} - Ax_i - f_i \right\|_Q^2$$

leads to the same estimate of (2.4) so long as $Q > 0$. See Appendix B for derivation of (2.4) and for proof of the above statement.

To express (2.4) in a recursive form, consider

$$\hat{A}_u(N+m) = \hat{A}_u(N) + \alpha(1) \tag{2.5}$$

where additional m vectors are incorporated together. Noting that terms linear in α vanishes in $J_N\left(\hat{A}_u(N) + \alpha\right)$ by the optimality of $\hat{A}_u(N)$, the optimal correction terms, $\hat{\alpha}$, is obtained from

$$\min_{\alpha} \operatorname{tr}\left\{\alpha A_k Z_{0,N+m-1} A_k^T \alpha^T + \alpha A_k \left(Z_{N,N+m} A_k^T \hat{A}_1(N)^T - W_{N,N+m}^T\right) + \left(-W_{N,N+m} + \hat{A}_u(N) A_k Z_{N,N+m}\right) A_k^T \alpha^T\right\}$$

where $W_{N,N+m}$ is defined similarly to $W_{0,N-1}$. Proceeding as before, the best correction term given the k^{th} block of m pieces of data is

$$\hat{\alpha}(k) = \left(W_{N+(k-1)m,N+km} - \hat{A}_u\left(N+(k-1)m\right) A_k Z_{N+(k-1)m,N+km}\right) \times A_k^T (A_k Z_{0,N+km-1} A_k^T)^{-1}$$

$$= \left[\sum_{i=N+(k-1)m}^{N+km-1} \left(y_{i+1} - f_i - \hat{A}_u\left(N+(k-1)m\right) A_k y_i\right) y_i^T\right] A_k^T (A_k Z_{0,N+km-1} A_k^T)^{-1} \tag{2.6}$$

Noisy Measurements

Now consider the same plant equation with noisy observations obtained as

$$y_i = x_i + \eta_i$$

where

$$E\,\eta_i = 0$$

$$E\,\eta_i \eta_j^T = R_i \delta_{ij}, \quad R_i > 0$$

Noting that

$$y_{i+1} = A\,y_i + \zeta_i + f_i$$

where

$$\zeta_i \triangleq \eta_{i+1} - A\eta_i$$

pick the criterion function to be given by

$$J_N = \sum_0^{N-1} \left\| y_{i+1} - A\,y_i - f_i \right\|^2$$

which leads to the best estimate of A_u given by a formula similar to (2.4) when y's are substituted for x's in matrices defined by (2.3)

$$\hat{A}_u(N) = W_{0,N-1} A_k^T (A_2 Z'_{0,N-1} A_k^T)^{-1}$$

where

$$W'_{0,N-1} \triangleq \sum_0^{N-1} (y_{i+1} - f_i) y_i^T$$

$$Z'_{0,N-1} \triangleq \sum_0^{N-1} y_i y_i^T$$

The recursive formula is obtained as

$$\hat{A}_u(N+km) = \hat{A}_u\Big(N+(k-1)m\Big) + \alpha(k)$$

where

$$\hat{\alpha}(k) = \left[\sum_{i=N+(k-1)m}^{N+km-1} \Big(y_{i+1} - f_i - \hat{A}_u\Big(N+(k-1)m\Big) A_k y_i\Big) y_i^T\right] \times A_k^T (A_k Z_{0,N+km-1} A_k^T)^{-1} \tag{2.7}$$

Since this estimate of $\hat{A}_u$ is a least square estimate, and since noises ζ are correlated, this estimate is not satisfactory unless the covariance matrices R's are very close to null matrices.

Because the covariance matrices for ζ depend on the unknown matrix A_u, the maximum likelihood estimate of A_u turns out to be rather complicated and will be treated in a separate paper.

Aggregation Method

Another scheme computationally simpler than that of (2.4) is next considered. Regarding A_k as an aggregation matrix,[9] define the aggregated state vector z at time i by

$$z_i = A_k x_i$$

Then the z-vector is described by the dynamic equation given by

$$z_{i+1} = F z_i + v_i$$

where

$$v_i = A_k f_i \, , \quad z_o = A_k x_o$$

and where F is an $(\ell \times \ell)$ matrix

$$F = A_k A_u \tag{2.8}^\dagger$$

Take the criterion function of identification to be given by

$$J_N = \sum_{i=0}^{N-1} \| z_{i+1} - F z_i - v_i \|^2$$

The optimal estimate of F, based on N observations, is given by

$$\hat{F}(N) = W_F(N)\, Z_F(N)^{-1}$$

provided $Z_F > 0$ where

---

† More generally, if it is possible to find an $(\ell \times n)$ matrix C such that

$$C A_u A_k = F C$$

then, z_i is defined by

$$z_i = C x_i$$

$$\begin{cases} Z_F(N) \triangleq \sum_0^{N-1} z_i z_i^T \\ W_F(N) \triangleq \sum_0^{N-1} (z_{i+1} - v_i) z_i^T \end{cases}$$

Note that Z_F and W_F are $\ell x \ell$ matrices. Since

$$W_F(N) = FZ_N$$

F is identified exactly if no disturbances are present. Given an estimate of F, (2.8) may be solved exactly for certain subclasses of A_u. Generally an estimate of A_u obtained as

$$\hat{A}_u(N) = A_k^+ W_F(N) Z_F(N)^{-1} \qquad (2.9)^{\dagger}$$

is in error, since

$$\hat{A}_u(N) = A_k^+ A_k A_u$$

$$A - \hat{A}(N) = \left(A_u - \hat{A}_u(N)\right) A_k = (I - A_k^+ A_k) A_u A_k$$

Thus, there will be an error in identifying A even if no random disturbances are presented for general A. For certain subclasses of A, however, A will be identified exactly and the identification procedure is computationally simpler. See Example 1 at the end of this paper.

3. IDENTIFICATION OF DETERMINISTIC SYSTEMS WITH CONSTRAINTS, II

Next, consider the case where A is constrained by (1.2a) and (1.2b) of Introduction. Then, as shown in Appendix C, A has the representation

$$A = DEC \qquad (3.1)$$

where D, E, C are $(n x \ell)$, $(\ell x \ell)$ and $(\ell x n)$ matrices respectively and are such that

$$CQ = 0_{(n,k)} \qquad (3.2a)$$

and

$$PD = 0_{(k,n)} \qquad (3.2b)$$

Now choose C subject to (3.2a). See Appendix D. This C is used to aggregate x into z

$$z_i = Cx_i$$

which satisfies the dynamic equation

$$z_{i+1} = Fz_i + g_i \qquad (3.3)$$

---

† The same estimate of A_u results if (2.8) is substituted into the expression of J_N and the resultant expression is minimized with respect to A_u. It is possible to put (2.9) into a recursive formula similar to (2.5) and (2.6).

where

$$g_i = Cf_i$$

and

$$F = CDE : \quad \ell x \ell \text{ matrix} \tag{3.4}$$

From the discussions in Section 2, it is seen that the matrix F can be identified exactly in the absence of random disturbances.

Having identified F exactly, (3.4) must be solved for D and E subject to (3.2a). Choosing C and D so that CD is nonsingular

$$E = (CD)^{-1}F$$

and

$$A = D(CD)^{-1}FC$$

As will be shown in Section 4, there is some intuitive appeal for the choice of C and D subject to

$$CD = I_\ell \tag{3.5}$$

then from (3.4) it is seen that E is identified directly as F.

From Appendix C and D the matrices C and D can be represented as

$$C = C_2(-Q_2Q_1^{-1}, I_\ell)$$

and

$$D = \begin{pmatrix} -P_1^{-1}P_2 \\ I_\ell \end{pmatrix} D_2$$

where C_2 and D_2 are ($\ell x \ell$) nonsimilar matrices. If the matrix D_2 is given by

$$D_2 = (Q_2Q_1^{-1}P_1^{-1}P_2 + I_\ell)^{-1} C_2^{-1}$$

then (3.5) is satisfied. Analogous to Equation (2.8), a recursive equation for identifying F in (3.4) is given by

$$\hat{F}(N+km) = \hat{F}\Big(N+(k-1)m\Big) + \left[\sum_{N+(k-1)m}^{N+km} \Big(z_{i+1} - \hat{F}\big(N+(k-1)m\big)\, z_i - g_i\Big)\, z_i^T\right] Z^{-1}_{0,N+km-1} \tag{3.6}$$

where $Z_{0,N+km-1}$ is as defined in (2.6).

Noisy Measurements

Suppose that the plant dynamics is described by $x_{i+1} = Ax_i + f_i$ and the aggregation matrix C results in a noisy aggregated vector $z_i = Cx_i + \eta_i$. Then $z_{i+1} = Fz_i + g_i - F\eta_i$. Discussions of this recursive scheme are quite analogous to those given in Section 2.

4. APPROXIMATE IDENTIFICATION, INTERPRETATIONS AND DISCUSSIONS

An approximate identification scheme of A is now discussed using the concept of aggregation and its converse, disaggregation concept. It is shown that this approximation method amounts to assuming that A satisfies the constraints (1.2a) and (1.2b) approximately. It was shown in Section 3 that if A is in the form of (3.1) and such that $(CD)^{-1}$ exists, then A can be identified as

$$A = D(CD)^{-1}FC \tag{4.1}$$

This method works whenever A is known to be of the form given by (3.2), independent of the fact that D and C may be constrained. In a general case, consider an aggregated system of S_1 governed by

$$z_{i+1} = Fz_i + g_i \tag{4.2}$$

where $FC = CA$ where C satisfies (3.2a). Now estimate x_i by

$$\hat{x}_i = Dz_i \tag{4.3}$$

The matrix D is called the disaggregation matrix. The choice of D is subject to the constraint Equation (3.2b). The above procedure generates an estimate $\hat{A}$ of A given by

$$\hat{A} = DFC \tag{4.4}$$

since $x_{i+1} = \hat{x}_{i+1} + e_{i+1} = DFCx_i + e_{i+1}$ where e_{i+1} is the estimation error $x_{i+1} - \hat{x}_{i+1}$.

The matrices D and C are to be chosen to minimize the identification error in some appropriate sense, subject to the constraint equations so that $\hat{A}$ becomes a member of the class $\hat{A}Q = 0_{n,k}$, $P\hat{A} = 0_{k,n}$. Equation (4.4) implies that the aggregation operation $z_i = Cx_i$ and the estimator of the form $\hat{x}_i = Dz_i$ are being used when A does not necessarily satisfy the structural constraint of Equations (1.2a) and (1.2b).

Practical significance of the above procedure is that an estimate of x obtained by (1) aggregate of x_j into z_j, (2) advance in time $z_{j+1} = Fz_j + v_i$, (3) estimate x_{j+1} by $\hat{x}_{j+1} = Dz_{j+1} = DFCx_j$ where $PD = 0_{(k,\ell)}$, $CQ = 0_{(\ell,k)}$ is equivalent to estimating x_{j+1} by x_j from $x_{j+1} = \hat{A}x_j$ subjected to the constraint on $\hat{A}$ given by $P\hat{A} = 0$, $\hat{A}Q = 0$. Thus for A with certain structural properties, the aggregation and disaggregation operations can be performed with no loss of information at a saving in the computational amounts. For other A's, the operations involve certain loss of information which may be tolerable in view of the lesser computational requirements.

Examples

1. Consider a (3x3) matrix A constrained by $AQ = 0$ where $Q^T = (1,1,1)$. Then $A = A_u A_k$ where $A_u = (a\rangle, b\rangle)$ where $\langle a = (a_1, a_2, a_3)$, $\langle b = (b_1, b_2, b_3)$ and where

$$A_k = \begin{pmatrix} -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$$

Then F is identified exactly by $A_k A_u$. The null space of A_k is spanned by (1,1,1). Hence A is identified exactly from F if $\Sigma_i a_i = \Sigma_i b_i = 0$.

2. The system of this example arises from a simplified model of an economic system where $A = A_1 + A_2$ and where A_1 is a known matrix, and A_2 is the matrix containing five unknown parameters and is given by $A_u = [O_{94} \vdots a\rangle\langle 1,1,1,1,1]$ where O_{94} is (9x4) null matrix and where $\langle a = (a_1, a_2, a_3, a_4, a_5, a_1, a_2, a_3, a_4)$. Then $A_u Q = 0$ where $Q^T = (I_4, O_{45})$. Thus $A_2 = A_u A_k$ where $A_u = a\rangle\langle 1,1,1,1,1$ and $A_k = (O_{54}, I_5)$. Since F identifies $a_1 \sim a_5$, the identification of F is enough to identify A_u.

REFERENCES

1. Fukao, T., "System Identification by Bayesian Learning Process," Proc. IFAC Tokyo Symposium, 137-146, August 1965.
2. Lee, R.C.K., Optimal Estimation, Identification and Control, MIT Press, Cambridge, Mass, 1964.
3. Ho, Y-C and B.H. Whalen, "An Approach to the Identification and Control of Linear Dynamic Systems with Unknown Parameters," IEEE Trans. Aut. Control AC-8, 255-6, No. 3, July 1963.
4. Ho, Y-C and R.C.K. Lee, "Identification of Linear Dynamic Systems," Inf. and Control 8, 93-110, 1965.
5. Aoki, M., Optimization of Stochastic Systems, Academic Press, New York, 1967.
6. Sworder, D.D., Optimal Adaptive Control Systems, Academic Press, New York, 1966.
7. Bellman, R. and R. Kalaba, "Quasilinearization and Nonlinear Boundary Value Problems," RAND Corp., Santa Monica, Calif. 1965.
8. Dvoretzky, A., "On Stochastic Approximations," Proc. 3rd Berkeley Symp. Math. Stat. and Probability, Vol. 1, 39-55, 1956.
9. Aoki, M., "Control of Large Scale Dynamic Systems by Aggregation," Proc. J.A.C.C., pp. 178-186, Univ. Penn., Philadelphia, June 1967.
10. Albert, A., and R.W. Sittler, "A Method for Computing Least Squares Estimators the Keep Up With Data," SIAM J. Control 3, 384-417, No. 3, 1965.
11. Fisher, W.D., "Optimal Aggregation in Multi-Equation Prediction Models," Econometrica 30, 744-769, No. 4, October 1962.

APPENDIX A

Factorization of A

Given an (nxn) matrix A subject to

$$AQ = 0_{(n,k)} \tag{A.1}$$

where Q is a known (nxk) matrix, write $Q^T = (Q_1^T, Q_2^T)$ where Q_1 is (ℓxk) and Q_2 is (kxk) matrices respectively and where it is assumed by suitable permutations that Q_2 is nonsingular. Write $A = (A_R \vdots A_L)$ where A_R is $(nx\ell)$- and A_L is (nxk)-matrix. From Equation (A.1) $A_RQ_1 + A_LQ_2 = 0$ or $A_L = -A_RQ_1Q_2^{-1}$ since Q_2 is nonsingular by assumption. Thus, $A = (A_R \vdots -A_RQ_1Q_2^{-1}) = A_uA_k$ where $A_u \triangleq A_R$ is the $(nx\ell)$ unknown matrix and where $A_k \triangleq (I_\ell \vdots -Q_1Q_2^{-1})$ is a known (ℓxn) matrix of rank ℓ.

By considering a partition of Q where Q_1 is now assumed to be a nonsingular (kxk) and Q_2 is (ℓxk) matrix, the same equation, (A.1) gives rise to a representation of $A = A_uA_k$ where $A_u = A_L \vdots (nx\ell)$ unknown matrix, and where $A_k = (-Q_2Q_1^{-1}, I_\ell)$: $(nx\ell)$ known matrix.

APPENDIX B

Minimization of J_N

Given

$$J_N = \mathrm{tr}\left\{ -AW^T_{0,N-1} - W_{0,N-1}A^T + A\,Z_{0,N-1}A^T \right\} + \text{const.}$$

where $w_i = x_{i+1} - f_i$ it is desired to minimize J_N with respect to A_u. The necessary condition for optimal A_u is given by

$$0 = \mathrm{tr}\left\{ -\Delta A_k W^T_{0,N-1} - W_{0,N-1}A_k^T\Delta^T + \Delta A_k Z_{0,N-1}A_k^T\hat{A}_u + \hat{A}_u A_k Z_{0,N-1}A_k^T\Delta^T \right\} \qquad \text{(B.1)}$$

for arbitrary $n \times \ell$ matrix Δ, or

$$\hat{A}_u A_k Z_{0,N-1} A_k^T = W_{0,N-1}A_k^T \qquad \text{(B.2)}$$

The same equation, (B.2) results if J_N is given by

$$J_N = \sum_0^{N-1} (w_i - Ax_i)^T Q (w_i - Ax_i) \quad \text{where } Q > 0.$$

APPENDIX C

The representation of A satisfying the constraint equations, Equations (1.2a) and (1.2b) is shown in a paper by Fischer[11] and included here for the sake of completeness.

Write A as $\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$ where A_{11} is $(k \times k)$ and A_{22} is $(\ell \times \ell)$. By suitable permutations one can write $P = (P_1 \vdots P_2)$ where P_1 is an $(k \times k)$ and P_2 is $(k \times \ell)$ matrix and where P_1 is nonsingular. Similarly, write $Q^T = (Q_1^T, Q_2^T)$ where Q_1 is an $(k \times k)$ and Q_2 is $(\ell \times k)$ matrix where Q_1 is nonsingular. Writing out the constraint equations and solving for submatrices of A, it is seen that A is in a form given by Equation (1.3c). The fact that A satisfies the constraints of Equation (1.2a) and (1.2b) is clear from the construction.

APPENDIX D

Existence of C and D Matrices

Write $C = (C_1 \vdots C_2)$ where C_1 and C_2 are $(\ell \times k)$ and $(\ell \times \ell)$ respectively, and where C_2 is chosen to be nonsingular.

From (3.2a) and Appendix C,

$$C_1 = -C_2 Q_2 Q_1^{-1} \qquad \text{(D.1)}$$

Write $D^T = (D_1^T, D_2^T)$ where D_1 and D_2 are $(k \times \ell)$ and $(\ell \times \ell)$ respectively. From the constraint (3.3a) $P_1D_1 + P_2D_2 = 0$ or

$$D_1 = -P_1^{-1} P_2 D_2 \qquad \text{(D.2)}$$

since P_1 is chosen to be nonsingular.

ADAPTIVE CONTROL WITHOUT COMPLETE PLANT IDENTIFICATION

N. K. SINHA
Department of Electrical Engineering
McMaster University
Hamilton, Ontario

M. J. IQLEEM
Atomic Energy of Canada Limited
Chalk River, Ontario

ABSTRACT

Adaptive control of a plant with changing parameters can be implemented without a complete identification of plant parameters if the cost function is measured continuously, and the controller parameters adjusted until the optimum is reached. A comparative study of the method of steepest descent, the method of conjugate gradients and the method of parallel tangents was made for a system based on the ITSE criterion. It was found that the method of parallel tangents required the least number of iterations.

INTRODUCTION

There are two basic philosophies underlying adaptive control. One is that the characteristics (transfer function or vector differential equation) of the plant to be controlled must be determined completely, and then the values of the parameters of an adjustable controller can be computed so that the performance is optimized for a given figure of merit (or cost function). The process of determining the dynamic characteristics of the plant is called "plant identification", and constitutes a major problem in the realization of an adaptive system of this type.

The other approach is based on the premise that a complete knowledge of the plant is not necessary for the optimum adjustment of the controller if the cost function is measured continuously, as in this case one can keep on adjusting the parameters of the controller until the optimum is reached. A very good example is of a pilot flying an airplane (1). The pilot is constantly identifying the plane's behaviour without doing so in terms of coefficients of a differential equation or transfer function. What he senses is simply the current performance of his plane, along with its current sensitivity to the control action. A number of "direct optimization" techniques, based on this philosophy, have been proposed in the literature (2), (3), (4), (5), (6). The most notable among these are the method of steepest descent, the method of conjugate gradients, and the method of parallel tangents. Considerable amount of work has been done on the application of these methods to the minimization of functions of several variables, and recently Box (7) has published a paper comparing their relative merits. It has been found that the method of conjugate gradients and the method of parallel tangents are very effective when the function to be minimized is in the form of a sum of polynomials of the adjustable parameters.

In applying these techniques to control systems, one is immediately faced with the difficulty that the cost function is usually in the form of an integral, and one of the most suitable performance criteria is the ITSE (integral of time and squared-error) criterion. But with the exception of Gibson's work (which considers only the method of steepest descent) (8), very little has been reported on the application of the direct optimization techniques to systems with practical performance criteria. As it would be very desirable to know how effective these methods are in such a case, the object of this paper is to investigate the comparative effectiveness of the three methods, (including some of their variants), in the optimization of a system based on the ITSE criterion. The result of this study would also tell how feasible is adaptive control without complete plant identification.

BRIEF REVIEW OF DIRECT OPTIMIZATION TECHNIQUES

A number of direct optimization techniques have been discussed in the literature (2), (3), (4), (5), (6), and in many cases, a number of modified schemes have been suggested. It is not the purpose of this paper to go into the details of these methods; as these are available in the references cited above. We shall only consider the basic ideas behind the methods. After a careful study of these methods, it was decided to adopt the following:

(a) The Method of Steepest Descent

This method is of historic significance and dates back to Cauchy. The basic concept is quite straight-forward, i.e., the direction of steepest descent at any point is defined by the negative of the gradient vector at that point. This involves determination of the partial derivatives of the cost function with respect to the variable parameters of the controller. In case the cost function is an integral, this determination of partial derivatives becomes a rather slow process. In practice, a discrete approximation of the path from the starting point to the minimum is obtained by proceeding along the direction of the gradient, assuming that the gradient does not change, until the lowest point on this line is found. The direction of the gradient is again computed, and steps taken from this point in the new direction. If the starting point happens to be on the side of a ridge of the response surface, the method may not progress well. A number of modifications have been proposed: the two most commonly employed are due to Booth (9) and Rosenbrock (10). As these modifications usually introduce only slight improvement, they have not been tried here.

(b) The Method of Conjugate Gradients

It was decided to use Fletcher and Reeves' adaptation (5) of the conjugate gradient methods of Hestenes and Strefel (11). In this method, one starts from the initial point in the direction of the steepest descent. However, after the position of the minimum on this line is obtained, the next step is taken in a direction which is a linear combination of the gradient at this point and the previous gradient. The following formulas define the conjugate gradient iterative procedure:

x_o = arbitrary starting point
$g_o = g(x_o)$ represents the gradient at x_o
$p_o = -g_o$ is the initial direction
x_{i+1} = position of the minimum of $F(x)$ on the line through x_i in the direction of p_i (1)
$g_{i+1} = g(x_{i+1})$ is the gradient at x_{i+1}
$B_i = \frac{gi^2 + 1}{g_i^2}$
$p_{i+1} = g_{i+1} + B_i p_i$

It has been claimed (5) that this procedure locates the minimum of a positive definite quadratic function of n variables in at most n iterations (apart from rounding off errors, etc.). For the general case, this procedure is iterative rather than n-step.

(c) The Method of Parallel Tangents

The method of parallel tangents or "PARTAN" as it is called, is based on certain global properties of ellipsoids. For criterion functions with concentric ellipsoidal contours, PARTAN will find the optimum exactly after a fixed, small number of iterations. But even when the contours are not precisely elliptical, the technique has desirable features.

The basic technique is most easily understood by considering a two-dimensional example. Suppose F (x_1, x_2) has elliptical contours and P_o and P_2 are any two points in the parameter plane. From P_2 progress is made in the direction parallel to the tangent to the contour at P_o until an extremum of F is found. Then the tangents at P_o and P_3 will be parallel and the centre of the system will be found at P_4, the extremal point on the line through P_o and P_3. Minima are determined on two parallel lines, and for elliptical contours these minima occur at mid-points of the chords. The theorem of parallel chords asserts that these minima are collinear with the centre of the ellipse. The approach can be extended to n-dimensions in which P_{2n-4} and P_{2n-1} have parallel tangents, and the extremum of F is found at P_{2n}.

The following variants of PARTAN were used.

(i) Steepest Descent PARTAN: From any point P_o proceed along a polygonal line, P_o P_2 P_3 P_4 ....., for which P_K is the minimum of F on the extended line joining it to the preceding point, P_{K-1}. At even-numbered points proceed in the direction of steepest descent. At odd-numbered points, P_{2K+1} proceed in the direction determined by the line joining P_{2K-2} and P_{2K+1}. Steepest descent PARTAN is in fact an n-dimensional generalization of the two dimensional procedure of two steepest descents followed by an acceleration step.

(ii) General PARTAN: From any point P_o progress is made along a polygonal line P_o P_2 P_3 P_4 ...., for which P_K is the minimum of F on the extended line joining it with the preceding point. Let π_i be the hyperplane tangent to the contour of F at P_i. The direction of P_o P_2 is arbitrary; P_2 P_3 is any direction parallel to π_o; thereafter, for $K = 1, 2, ..., n-1$, P_{2K+2} is collinear with P_{2K-2} and P_{2K+1}, and for $K = 2, 3, ..., n-1$, P_{2K} P_{2K+1} is parallel to $\pi_o, \pi_2, ..., \pi_{2K-2}$.

(iii) Scale Invariant PARTAN: This is a special case of the general PARTAN described above. In this case the arbitrary directions are resolved as follows: at P_o only x_1 is changed; at P_2, x_1 and x_2 are changed; at P_4, x_1, x_2 and x_3 are changed; and so on.

An elaborate account of the method of parallel tangents is also given by Wilde (2). Buehler, Shah and Kempthorne (6) have shown that if F is quadratic, the minimum is reached at P_{2n} or sooner, n being the number of dimensions. In non-quadratic case more iterations are generally needed.

APPLICATION TO CONTROL SYSTEMS

To study the behaviours of the various methods, a system suggested by Gibson (8) was considered. This system is shown in Figure 1. With the ITSE performance criterion, this system has closed contours of constant figure-of-merit curves in the parameter plane of ω_1 and ω_2, and has been recommended by Gibson as suitable for optimization by gradient methods. Gibson has also pointed out the difficulties with systems which do not have closed contours of constant figure-of-merit curves in the parameter plane.

Starting with arbitrary initial values of ω_1 and ω_2, the various methods were used to determine the number of iterations required to reach the optimum in each case. The overall transfer function of the system was resolved into six first order simultaneous differential equations, which were solved on an IBM7040 computer using the Runge-Kutta method. The function $F(\omega_1, \omega_2)$ was obtained from these solutions, and the partial derivatives were calculated numerically by local examination of the response surface.

Some of the results are tabulated below: (12)

(a) Starting Point $\omega_1 = 0.31$, $\omega_2 = 10.0$

(i) Steepest Descent:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.31	10.10	0.022460
2	0.29089	10.51047	0.022039
3	0.31639	10.54910	0.021587
4	0.30369	10.68751	0.021475
5	0.41639	10.74097	0.021320
6	0.30369	10.91876	0.021211
7	0.31640	10.95692	0.021047
8	0.30370	11.26434	0.020880
9	0.32280	11.29986	0.020675
10	0.31650	11.34355	0.020625
11	0.34200	12.09583	0.020165
12	0.32289	12.12183	0.019951
13	0.34840	12.98325	0.019617
14	0.32930	13.00912	0.019423
15	0.33880	13.14385	0.019380
16	0.33249	13.15878	0.019342
17	0.33589	13.20049	0.019341

(ii) Conjugate Gradients:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.31	10.0	0.022460
2	0.29089	10.51047	0.022039
3	0.31200	10.54847	0.021588
4	0.30370	10.68290	0.021476
5	0.31640	10.73879	0.021323
6	0.30370	10.91204	0.021219
7	0.31640	10.94989	0.021056
8	0.30370	11.23972	0.020901
9	0.32279	11.27576	0.020701
10	0.31330	11.34033	0.020650
11	0.32599	11.42836	0.020560
12	0.32599	11.47411	0.020500
13	0.32920	11.64516	0.020368
14	0.31970	11.68465	0.020300
15	0.37079	14.21058	0.019560
16	0.34530	14.22722	0.018997

(iii) Iterated Steepest Descent PARTAN:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.31	10.0	0.022460
2	0.29089	10.51047	0.022040
3	0.36760	17.68500	0.018744
4	0.36290	17.68680	0.018733
5	0.36280	17.01969	0.018720
6	0.35970	17.02081	0.018712

(iv) General PARTAN:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.31	10.0	0.022460
2	0.29089	10.51047	0.022040
3	0.31640	10.54909	0.021587
4	0.30370	10.68752	0.021475
5	0.31639	10.74097	0.021320
6	0.30370	10.91876	0.021211
7	0.31640	10.95692	0.021047
8	0.30369	11.26435	0.020880
9	0.32279	11.29986	0.020675
10	0.31650	11.34355	0.020625
11	0.34199	12.08583	0.020165
12	0.32289	12.12182	0.019951
13	0.34839	12.98325	0.019617
14	0.32929	13.00912	0.019423
15	0.33879	13..4385	0.019380
16	0.33249	13.15878	0.019340
17	0.33569	13.23326	0.019319

(v) Iterated Scale Invariant PARTAN:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.31	10.0	0.022460
2	0.30929	10.00000	0.022451
3	0.29660	11.18796	0.021170
4	0.29630	12.51777	0.020655
5	0.33460	12.51777	0.019713
6	0.32829	12.51777	0.012684
7	0.32199	13.87104	0.019311
8	0.32154	14.40805	0.019300
9	0.34745	14.40805	0.018960
10	0.34734	16.54398	0.018786
11	0.34725	16.12352	0.018776
12	0.35355	16.12352	0.018724

(b) Starting Point $\omega_1 = 0.41$, $\omega_2 = 20.0$

(i) Steepest Descent:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.41	20.0	0.019695
2	0.37169	20.00380	0.018940
3	0.37639	19.99561	0.018920
4	0.37230	19.99545	0.018918
5	0.37609	19.99512	0.018913

(ii) Conjugate Gradients:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.41	20.0	0.019695
2	0.37169	20.00380	0.018940
3	0.37640	19.99561	0.018922
4	0.37630	19.99545	0.018921

(iii) Iterated Steepest Descent PARTAN:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.41	20.0	0.019695
2	0.37170	20.00380	0. 018940
3	0.37160	17.48280	0.018787
4	0.36540	17.81840	0.018740

(iv) General PARTAN:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.41	20.0	0.019695
2	0.37170	20.00381	0.018940
3	0.37639	19.99561	0.018922
4	0.37629	19.99545	0.018921

(v) Iterated Scale Invariant PARTAN:

Number of Iterations	ω_1	ω_2	$F(\omega_1, \omega_2)$
1	0.41	20.0	0.019695
2	0.37170	20.0	0.018940
3	0.317170	17.48247	0.018790
4	0.37189	17.81199	0.018780

The comparison of the results obtained by the different methods in two cases is shown in Figures 2 and 3.

CONCLUSION

Implementation of adaptive control in situations where plant identification is either inconvenient or undesirable, has two important aspects. One is the figure of merit or criterion by which the system measures its performance with a view to optimize it. The other is the determination of the method for adjusting the parameters of the controller to affect this performance optimization which is most efficient and fast. In the past few years several optimization techniques have been developed, but very little work has been done to apply these to practical integral-type criteria preferred with control systems. In this paper three optimizing techniques have been examined and their relative efficiency compared when applied to two-parameter optimization of a sixth-order system based on the ITSE criterion. These results indicate that the steepest descent variant of the method of parallel tangents is the most efficient in the cases considered, while the method of conjugate gradients is only slightly superior to the method of steepest descent.

As a result of this study, it appears that in a number of practical cases, adaptive control may be implemented in a very economical manner by use of suitable logic circuits and some operational units. On the other hand, the approach based on identification may often require expensive computing equipment.

In a recent paper (13), Lasdon, Mitter and Waren have applied the conjugate gradient method to minimization of quadratic performance criteria in optimal control systems. They have proved some theorems in connection with the convergence of the method for such criteria, and given some examples showing the superiority of this method over the method of steepest descent. The case of non-quadratic functions still remains unresolved.

ACKNOWLEDGEMENT

This work was partially supported by the Government of Canada under the Colombo Plan, which is gratefully acknowledged.

REFERENCES

1. J. Zaborsky and W.L. Humphrey: "Control Without Model or Plant Identification", IEEE Transactions on Applications and Industry, November 1964.
2. D.J. Wilde: "Optimum Seeking Methods", Prentice-Hall Inc., 1964.
3. P.E. Fleischer: "Optimization Techniques in System Design", Chapter 6 of "System Analysis by Digital Computer", edited by F.F. Kuo and J.F. Kaiser, John Wiley and Sons, Inc., 1966.
4. R. Fletcher and M.J.D. Powell: "A Rapidly Convergent Descent Method for Minimization", The Computer Journal, Vol. 6, 1963, pp.163-168.
5. R. Fletcher and C.M. Reeves: "Function Minimization by Conjugate Gradients", British Computer Journal, July 1964.
6. R.J. Buehler, B.V. Shah and O. Kempthorne: "Method of Parallel Tangents", Chemical Engineering Progress Symposium Series, Vol. 60,1964
7. M.J. Box: " A Comparision of Several Current Optimization Methods and the use of Transformations in Constrained Problems", The Computer Journal, Vol. 9, pp. 67-78, May 1966.
8. J.E. Gibson: "Self-Optimizing or Adaptive Control Systems", Proceedings of the First International Congress of the International Federation of Automatic Control, Moscow, 1960.
9. A.D. Booth: "Numerical Methods", Butterworths, 1957.
10. H.H. Rosenbrock and C. Storey: "Computational Techniques for Chemical Engineers", Pergamon Press, 1965.
11. M.R. Hestenes and E. Steifel: "Method of Conjugate Gradients for Solving Linear Systems", J. Res. N.B.S., Vol. 49, 1952.
12. M.J. Iqleem: "On the Feasibility of Adaptive Control Without Identification", M.Eng., Thesis, McMaster University, February, 1967.
13. L.S. Lasdon, S.K. Mitter and A.D. Waren: "The Conjugate Gradient Method for Optimal Control Problems", IEEE Transactions on Automatic Control, April, 1967.

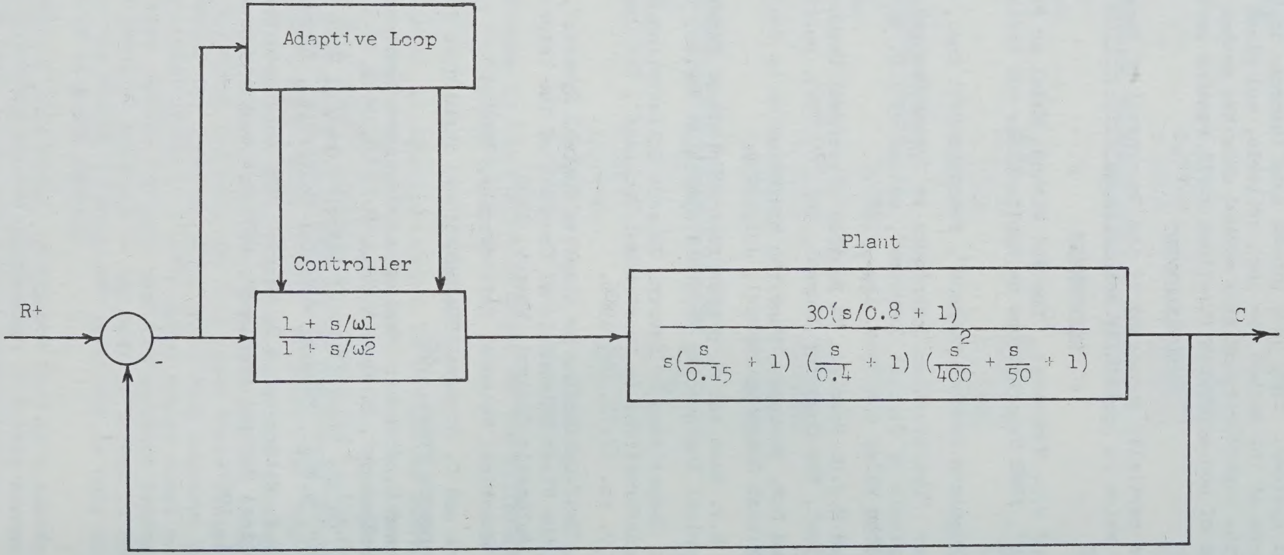

Adaptive Loop
Controller
$\frac{1 + s/\omega 1}{1 + s/\omega 2}$
Plant
$\frac{30(s/0.8 + 1)}{s(\frac{s}{0.15} + 1)(\frac{s}{0.4} + 1)(\frac{s^2}{400} + \frac{s}{50} + 1)}$
R+
-
C

Figure 1: Gibson's System

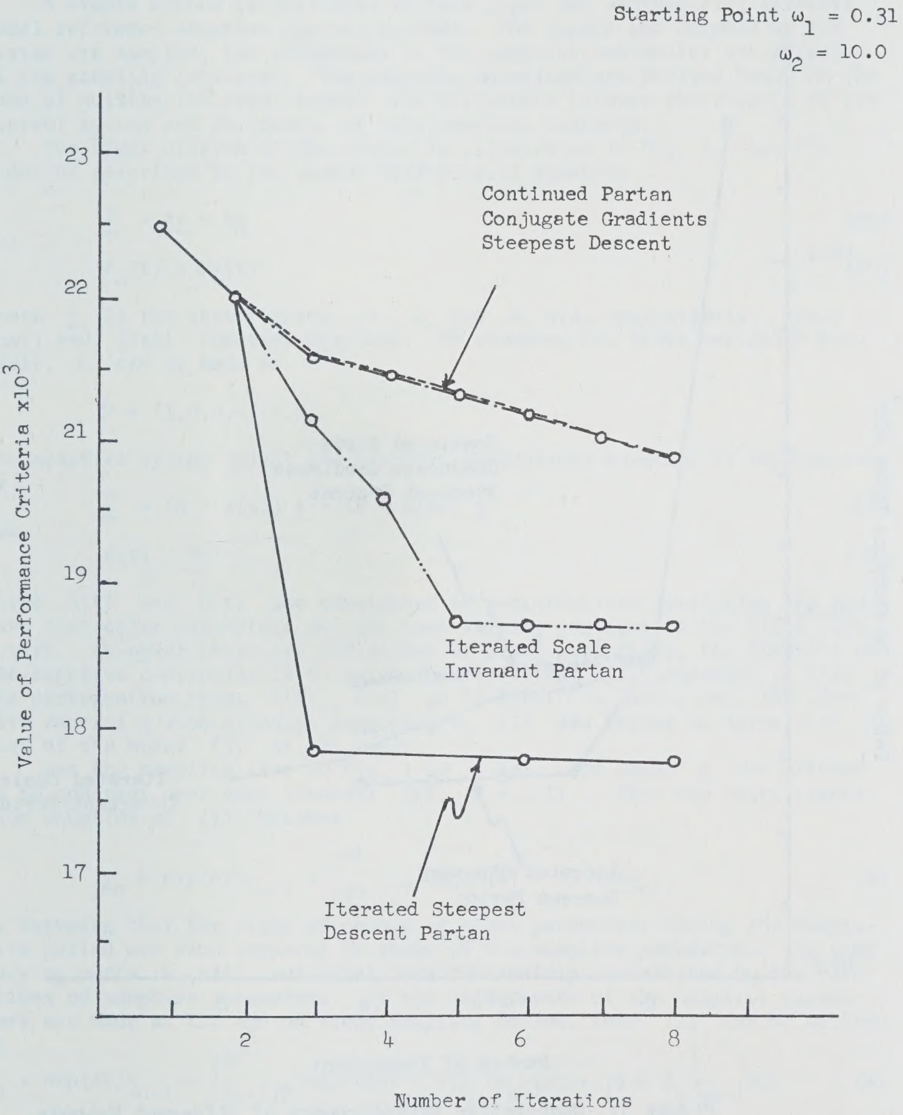


Figure 2: Comparative Effectiveness of Different Methods

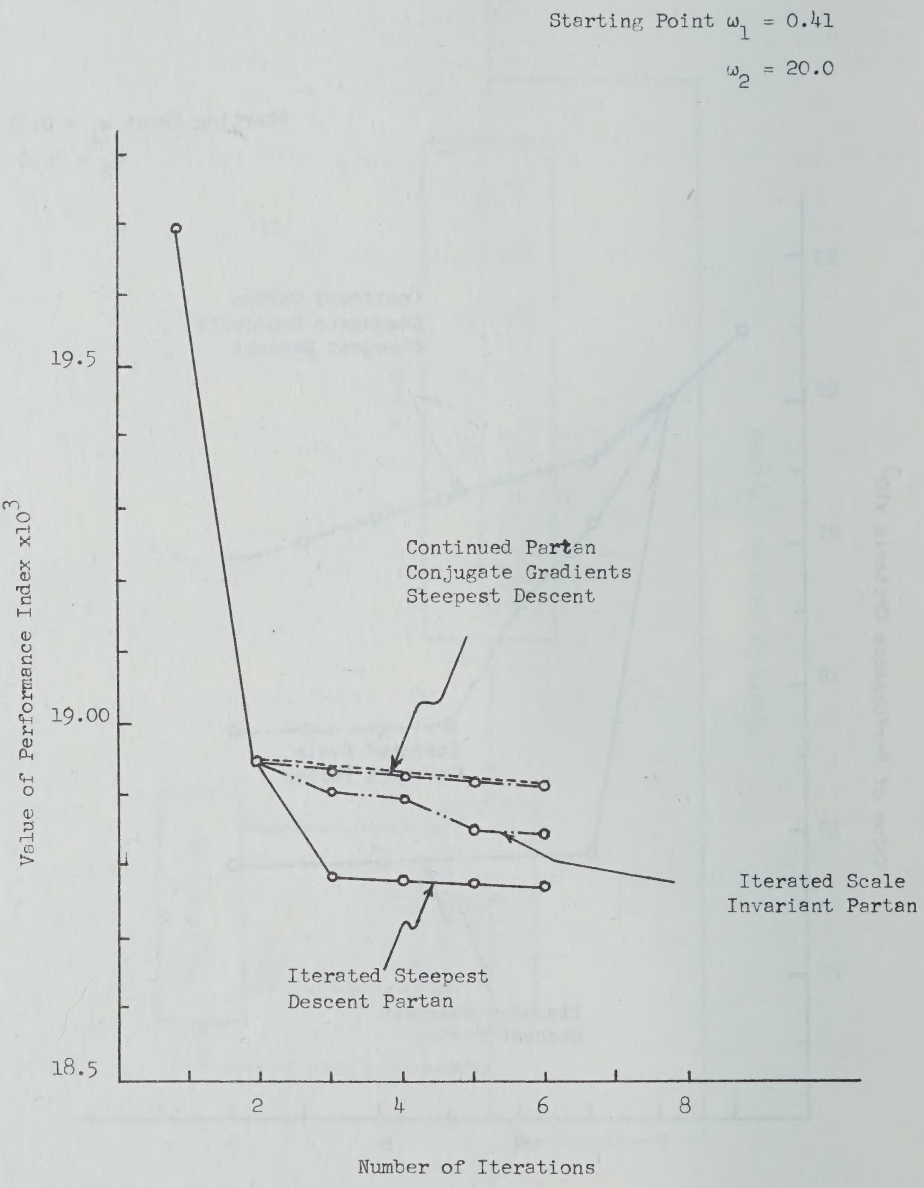


Figure 3: Comparative Effectiveness of Different Methods

A TECHNIQUE FOR THE SYNTHESIS OF DISCRETE ADAPTIVE CONTROL SYSTEMS

VALLOBH VIMOLVANICH & TIEN C. HSIA
Department of Electrical Engineering, University of California, Davis, California

A simple method is developed in this paper for synthesizing discrete model reference adaptive control systems. The inputs and outputs of the system are sampled, the parameters in the adaptive controller are adjusted at the sampling instances. The adaption equations are derived based on the idea of nulling the error signal, the difference between the outputs of the control system and the model, at each sampling instance.

The block diagram of the system is illustrated in Fig. 1. Let the model be described by the vector differential equation

$$\dot{\underline{y}} = A\underline{y} + B\underline{u} \tag{1a}$$

and

$$c_d(t) = M\underline{y}(t) \tag{1b}$$

where $\underline{y}$ is the state vector, A, B and M are, respectively, $(n\mathrm{x}n)$, $(n\mathrm{x}r)$ and $(1\mathrm{x}n)$ constant matrices. By choosing the state variables properly, M can be made as

$$M = [1,0,0,\text{----},0]$$

The adaptive system (plant and adaptive controller) equation is represented by

$$\dot{\underline{x}} = (A + A(t))\,\underline{x} + (B + B(t))\,\underline{u} \tag{2a}$$

and

$$c(t) = M\underline{x}(t) \tag{2b}$$

where $A(t)$ and $B(t)$ are considered as perturbations containing the adaptive controller parameters and the time-varying portions of the plant parameters. Whenever there are variations of plant parameters, the function of the adaptive controller is to automatically adjust its parameters so that the perturbation terms $A(t)$, $B(t)$ go to zero. In such a way, the adaptive control system dynamics described by (2) are forced to agree with that of the model (1) at all times.

Let the sampling time in Fig. 1 be T sec. The input $\underline{u}$ are assumed to be constant over each interval $(kT, (k+1)T)$. Then the state transition equation of (1) becomes

$$\underline{y}_n = \exp(AT)\underline{y}_{n-1} + \int_{(n-1)T}^{nT} \exp(A(nT - s))ds B\underline{u}_{n-1} \tag{3}$$

By assuming that the rates of change of plant parameters during the adaptation period are slow compared to those of the adaptive parameters, the time varying parts in $A(t)$ and $B(t)$ are then mainly contributed by the variations of adaptive parameters. If the adjustments of the adaptive parameters are made at the end of every sampling period, then (2) can be written as

$$\underline{x}_n = \exp(AT)\underline{x}_{n-1} + \int_{(n-1)T}^{nT} \exp(A(nT - s))\,\{A_n\underline{x}(s) + (B + B_n)\underline{u}_{n-1}\}ds \tag{4}$$

For small T,

$$\underline{x}_n \simeq \exp(AT)\underline{x}_{n-1} + \int_{(n-1)T}^{nT} \exp(A(nT - s))ds B\underline{u}_{n-1} + \exp(AT)\ \{A_n \int_{(n-1)T}^{nT} \underline{x}(s)ds$$
$$+ TB_n\underline{u}_{n-1}\} \qquad (5)$$

The adaption procedure proposed here is as follows:

The adjustments of adaptive parameters ΔA_n and ΔB_n, are determined from the condition that when ΔA_n and ΔB_n are added to A_n and B_n, the ideal value of $\underline{x}_n$, $\hat{\underline{x}}_n$, is equal to $\underline{y}_n$, i.e.,

$$\hat{\underline{x}}_n = \underline{x}_n + \exp(AT)\ \{\Delta A_n \int_{(n-1)T}^{nT} \underline{x}(s)ds + T\Delta B_n\underline{u}_{n-1}\} = \underline{y}_n \qquad (6)$$

By using linear approximation, (6) can be written as

$$\underline{y}_n - \underline{x}_n = T\exp(AT)\ \{\tfrac{1}{2}\Delta A_n(\underline{x}_n + \underline{x}_{n-1}) + \Delta B_n\underline{u}_{n-1}\} \qquad (7)$$

Since

$$e_n = M(\underline{y}_n - \underline{x}_n)$$

Therefore,

$$e_n = TM\exp(AT)\ \{\tfrac{1}{2}\Delta A_n(\underline{x}_n + \underline{x}_{n-1}) + \Delta B_n\underline{u}_{n-1}\}$$
$$= \tfrac{1}{2}\Sigma\Sigma k_j\,\delta a_{jk}(x_{k,n} + x_{k,n-1}) + \Sigma\Sigma k_p\,\delta b_{pq}u_{q,n-1} \qquad (8)$$

(8) is satisfied if

$$\delta a_{jk} = k_{jk}e_n/(x_{k,n} + x_{k,n-1})$$
$$\delta b_{pq} = k_{pq}e_n/u_{q,n-1} \qquad (9)$$

The adaptive equation is given by (9). The adaptive loop gains k_{jk} and k_{pq}, which are positive constants, can be suitably chosen for fast convergance.

To implement (9), it is necessary that the system state variables $\underline{x}$ are measurable. However, in the case where the transfer functions of the plant and model have no numerator dynamics, $\underline{x}$ could be obtained by taking successive derivatives of the system output c.

It is seen that this technique has the distinct advantage of being extremely simple to implement. The results of various examples showed that the adaptive procedures always converge rapidly.

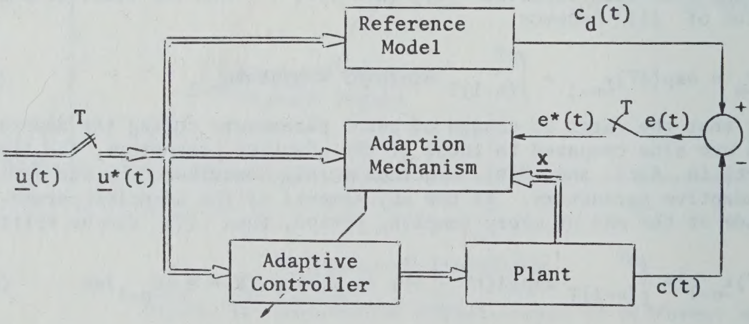


FIG. 1 DISCRETE MODEL REFERENCE ADAPTIVE CONTROL SYSTEM.

APPLICATION OF MATHEMATICAL PROGRAMMING TO THE DESIGN OF OPTIMAL CONTROL SYSTEMS

B. C. KUO
Dept. of Electrical Engineering
University of Illinois
Urbana, Illinois

D. TABAK
General Electric Company
Missile & Space Division
Philadelphia, Pennsylvania

ABSTRACT

The problem of applying the various computational methods of Mathematical Programming to the Design of Optimal Control Systems, is discussed. A general case of nonlinear, nonautonomous, state equations subject to inequality constraints on both state and control variables, is considered. Both continuous and discrete time systems are investigated. In case of discrete time systems, the sampling intervals are assumed generally unequal and aperiodic, with inequality constraints imposed upon them.

Systems like these, impose considerable computational difficulties when treated by the Maximum Principle or Dynamic Programming. Using Mathematical Programming one may simplify a wide class of those computational problems.

Several examples of applying Mathematical Programming to particular control problems, are presented.

INTRODUCTION

The methods applied in the design of optimal control systems are usually the maximum principle[5] and dynamic programming.[6] When using the maximum principle, one is faced with a solution of a two-point boundary value problem, the difficulties of which are well known, particularly in nonlinear and state-space constrained cases. Using dynamic programming, one has the so called "dimensionality curse," i.e.; requirements of computer fast memories of excessive sizes.[6]

Using mathematical programming, one avoids solving a two-point boundary value problem. The memory requirements are smaller than that in dynamic programming, and there is no problem in posing any kind of additional constraints.

A NONLINEAR PROGRAMMING PROBLEM

The most general case of Mathematical Programming is the Nonlinear Programming (NLP) problem.[1,2] It consists of extremizing a nonlinear function:

$$J = f(\underline{x}) \qquad (1)$$

where $\underline{x}$ is a n-vector of the unknown variables $x_1 \ldots x_n$, subject to a set of inequality

$$\underline{g}(\underline{x}) \geq \underline{0} \qquad (2)$$

and equality

$$\underline{h}(\underline{x}) = 0 \qquad (3)$$

constraints. There exist various computational methods of solving problems of this kind.[2] The method used in this paper is the method of Sequential Unconstrained Minimization Technique (SUMT).[3,4]

In the case when f, $\underline{g}$ and $\underline{h}$ are linear, one would have a case of Linear Programming (LP) and if $\underline{g}$, $\underline{h}$ are linear and f quadratic, it is usually called Quadratic Programming (QP).

THE GENERAL CONTROL PROBLEM

In the general control problem under consideration, the system is subject to a set of nonlinear, nonautonomous state equations:

$$\dot{\underline{y}} = \underline{f}(\underline{y},\underline{u},t) \qquad (4)$$

where $\underline{y}$ is the n-state and $\underline{u}$ the m-control vectors. At anytime the system is subject to a set of generally nonlinear constraints:

$$\underline{g}(\underline{y},\underline{u}) \geq 0$$

$$\underline{h}(\underline{y},\underline{u}) \geq 0$$

$$|u_i| \leq U; \quad i=1,\ldots,m \qquad (5)$$

as well as to a terminal constraint, which describes the target set,

$$G(\underline{y}_f) \geq 0 \qquad (6)$$

where $\underline{y}_f = \underline{y}_f(t_f)$. t_f may be fixed or variable. A general performance index to be extremized could be written as:

$$J = \int_{t_0}^{t_f} F(\underline{y},\underline{u},t)dt + H(\underline{y}_f) \qquad (7)$$

In particular, for H = 0 and F = 1, it becomes the minimal-time problem. Two different cases involving this problem should be noted.

(a) The closed--form solution of the state equations (4) is known:

$$\underline{y}(t) = \underline{\phi}[t,t_0,\underline{u}(t_0),\underline{y}(t_0)] \qquad (8)$$

where t_0 is the initial and t the current time, and $\underline{\phi}$ is an n-vector.

In this case the state equations could be written between discrete time instants t_i, $i = 0,1,\ldots,N$ as

$$\underline{y}(i) - \underline{\phi}[T_i,\underline{u}(i-1),\underline{y}(i-1)] = 0 \qquad (9)$$

where

$$\underline{y}(i) = y(t_i)$$

$$T_i = t_i - t_{i-1}$$

it may serve as a current variable of the problem. Equation (9) could also represent a basically discrete--time system described by difference equations, with $\underline{u}(i-1)$ being constant in the period $t_{i-1} \leq t < t_i$.

(b) The closed form solution of the state equations is unknown. The equations in (4) are rewritten now in difference form:

$$\underline{y}(i+1) - \underline{y}(i) = T_{i+1}\underline{f}[\underline{y}(i),\underline{u}(i),t_i] i = 0,1,\ldots,N-1 \qquad (10)$$

This is, of course, an approximation, which may not work for systems with very fast changing state variables. On the other hand, if the time change of the state variables is relatively slow, one may obtain a desired degree of precision by limiting the discrete time intervals from above:

$$T_i \leq T_{max}, \quad i = 1,\ldots,N \qquad (11)$$

One may also use a more elaborate difference scheme which could increase the accuracy. In some sampled--data systems, for practical reasons, one may have in addition a minimal sampling period restriction.

$$T_i \geq T_{min}, \qquad i = 1,\ldots,N \qquad (12)$$

APPLICATION OF NONLINEAR PROGRAMMING AND EXAMPLES

To apply NLP, in both cases, the N(1+m+n) variables of the problem are assigned in the following manner:

T_1	x(1)
⋮	⋮
T_N	x(N)
$u_1(0)$	x(N+1)
⋮	⋮
$u_m(0)$	x(N+m)
$u_1(1)$	x(N+m+1)
⋮	⋮
$u_m(1)$	x(N+2m)
⋮	⋮
$u_1(N-1)$	x[N+(N-1)m+1]
⋮	⋮
$u_m(N-1)$	x(N+Nm)
$y_1(1)$	x(N+Nm+1)
⋮	⋮
$y_n(1)$	x(N+Nm+n)
⋮	⋮
$y_1(N)$	x[N+Nm+(N-1)n+1]
⋮	⋮
$y_n(N)$	x(N+Nm+Nn)

These variables are appropriately substituted for all the constraints and the performance index.

The integral part of the performance index (7) is to be respecified as a summation.

$$J = \sum_{i=1}^{N} F_i[\underline{y}(i),\underline{u}(i-1)] + H[\underline{y}(N)] \qquad (13)$$

Now, one has a regular NLP problem, with (13) as a performance index, (5), (11), (12) as inequality and the state equations (9) or (10) as equality constraints.

These methods have been applied to several problems of particular applications as well as to general systems. In many cases, fast solutions have been obtained where one could not say the same if other methods were used.

Example 1.

Consider the following nonlinear system whose state equations are:

$$\dot{x}_1 = -(a+bu)x_1 + cx_2 + du$$

$$\dot{x}_2 = \qquad - x_2 + u \tag{14}$$

where

x_1, x_2 = state variables,

u = control variable,

a, b, c, d = constants.

The nonlinearity is introduced by the term ux_1 in the first of the state equations. The initial state $\underline{x}(0) = \underline{x}_0$ is given. The problem is to bring the system to the target line:

$$\frac{x_1}{x_2} = K \tag{15}$$

is minimal time.

To apply the NLP, time is divided into N intervals $T_i (i=1,\ldots,N)$ which are to be determined. The performance index is

$$\text{Minimize} \qquad J = \sum_{i=1}^{N} T_i \tag{16}$$

The state equations (14) and the terminal condition (15) are now rewritten in difference form:

$$x_1(i+1)-x_1(i) = [-(a+b\ u(i)) + cx_2(i) + d\ u(i)]T_{i+1} \tag{17}$$

$$x_2(i+1)-x_2(i) = [-x_2(i) + u(i)]T_{i+1} \qquad i = 0,1,\ldots,N-1$$

$$x_1(N) - K\ x_2(N) = 0 \tag{18}$$

Now one has a NLP problem with (16) as an objective function, and (17), (18) as equality constraints. In addition, the following inequality constraints are posed:

$$x_1(i) \le x_{max} \; ; \qquad i = 1,\ldots,N$$

$$u(i) \le u_{max} \; ; \qquad i = 0,1,\ldots,N-1 \tag{19}$$

The following data are used:

$a = .724$

$b = 9.47$

$c = 9.87$

$d = .324$

$x_1(0) = x_2(0) = 1.$

$x_{max} = 5.$

$u_{max} = 1.$

$K = 13.6$

For N=12, the minimal time is: .8055

For N=20, the minimal time is: .8062 a difference of about .09%

From the closeness of the results one may assume that the error introduced by the discretization of the state equations is tolerable.

Example 2. A Discrete-Time Nonlinear Nonautonomous System

The general formulation of the discrete-time control problem under consideration is as follows:

$$\text{Minimize} \quad J = \sum_{i=1}^{N} F_i[\underline{y}(i),\underline{u}(i-1)] \tag{20}$$

Subject to constraints:

$$\underline{g}[\underline{y}(i),\underline{u}(i)] \geq \underline{0}; \qquad i = 1,\ldots,N \tag{21}$$

and state difference equations:

$$\underline{y}(i+1)-\underline{y}(i) = \underline{f}_i[\underline{y}(i),\underline{u}(i)] = 0 \qquad i = 0,1,\ldots,N-1 \tag{22}$$

The initial state vector is specified:

$$\underline{y}(0) = \underline{y}_0 \tag{23}$$

In the example solved, a second-order system with one control variable was chosen. The system is described by the following difference state equations:

$$\begin{aligned} y_1(i+1)-y_1(i) - T_{i+1}[-y_1^2(i) + y_2(i)+u(i)] &= 0 \\ y_2(i+1)-y_2(i) - T_{i+1}y_1(i) &= 0 \\ i = 0,1,\ldots,N-1 & \end{aligned} \tag{24}$$

where, $y_j(i) = j^{\underline{th}}$ component of the state vector at the discrete time t_i.

$$T_j = t_j-t_{j-1}$$

The initial state of the system was:

$$\underline{y}(0) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

The problem was to bring the state into the target area:

$$[y_1(N)-10]^2 + y_2^2(N) - 1 \leq 0 \tag{25}$$

so that

$$J = \sum_{i=1}^{N} [y_1^2(i)+y_2^2(i)+.1\ u^2(i-1)] \tag{26}$$

is minimized.

The control variable was assumed bounded:

$$0 \leq u(i) \leq 1 \qquad i = 0,1,\ldots,N-1 \tag{27}$$

and the state variables, non-negative:

$$y_j(i) \geq 0 \qquad \begin{matrix} i = 1,\ldots,N \\ j = 1,2 \end{matrix} \tag{28}$$

As one may see, the target area represents a circle of radius 1 around the point (10,0). Two computational examples were solved. In both cases the number of sampling periods was chosen to be N=12. The difference between them is that in the first, the total time

$$T = \sum_{i=1}^{N} T_i$$

was assumed free, while in the second one, T was fixed to: T = 10.

The results of the free time example were as follows:

i	T_i sec	t_i sec	u(i-1)	$y_1(i)$	$y_2(i)$
1	.00291	.00291	.0553	.00265	.964
2	.00196	.00487	.5000	.00375	.930
3	.00176	.00663	.5000	.00434	.898
4	.00169	.00832	.0466	.00467	.867
5	.00164	.00996	.0435	.00479	.839
6	.00162	.01158	.0419	.00473	.812
7	.00159	.01317	.0406	.00448	.786
8	.00156	.01473	.0395	.00402	.762
9	.00151	.01624	.0383	.00330	.740
10	.00144	.01768	.0370	.00218	.719
11	.00130	.01898	.0356	.00012	.699
12	13.18224	13.20122	.0338	9.26978	.683

The minimal performance index turned out to be:

$$J_{min} = 93.86$$

The time variation of y_1, y_2, u is presented in Figure 1.

For the fixed time case, the following results were obtained:

i	T_i sec	t_i sec	u(i-1)	$y_1(i)$	$y_2(i)$
1	.00146	.00146	.404	.00205	.973
2	.07106	.07252	.500	.14261	.948
3	.00042	.07294	.500	.12606	.925
4	.00043	.07337	.992	.10976	.903
5	.00044	.07381	.023	.09369	.884
6	.00046	.07427	.023	.07781	.866
7	.00047	.07474	.023	.06208	.850
8	.00048	.07522	.022	.04648	.832
9	.00048	.07570	.022	.03099	.822
10	.00049	.07619	.022	.01555	.811
11	.00050	.07669	.022	.00012	.801
12	9.92503	10.00172	.022	9.36986	.776

$$J_{min} = 96.91$$

As one may see, a smaller performance index was obtained in the free time case. The results of the second example are presented graphically in Figure 2.

In both cases, the time variation of all the variables is similar. As one may see from Figures 1 and 2, most of the control effort is made during the first 3,4 discrete intervals. After that, a very small control signal is applied. y_1 obtains an overshoot and an undershoot to a very small value, and then increases directly to its target value in one long sampling interval. y_2 decreases very slowly during the process down to about 68% to 77% of its initial value.

In both cases, the nonlinear programming problem had 48 variables, and 37 constraints. The computation time for each problem was around 8 minutes on the IBM 7094.

REFERENCES

1. G. Hadley, "Nonlinear and Dynamic Programming," Addison Wesley, 1964.

2. G. Zontindijk, "Nonlinear Programming: A Numerical Survey," J. SIAM Control, 4, pp. 194-210, 1966.

3. A. V. Fiacco, G. P. McCormick, "The SUMT for Nonlinear Programming," Management Science, 10, pp. 360-366, 1964.

4. A. V. Fiacco, G. P. McCormick, "Extensions of SUMT for Nonlinear Programming: Equality Constraints and Extrapolation," Management Science, 11, pp. 816-828, 1966.

5. L. S. Pontriagin, et al., "The Mathematical Theory of Optimal Processes," John Wiley, N. Y., 1962.

6. R. Bellman, "Dynamic Programming," Princeton, 1957.

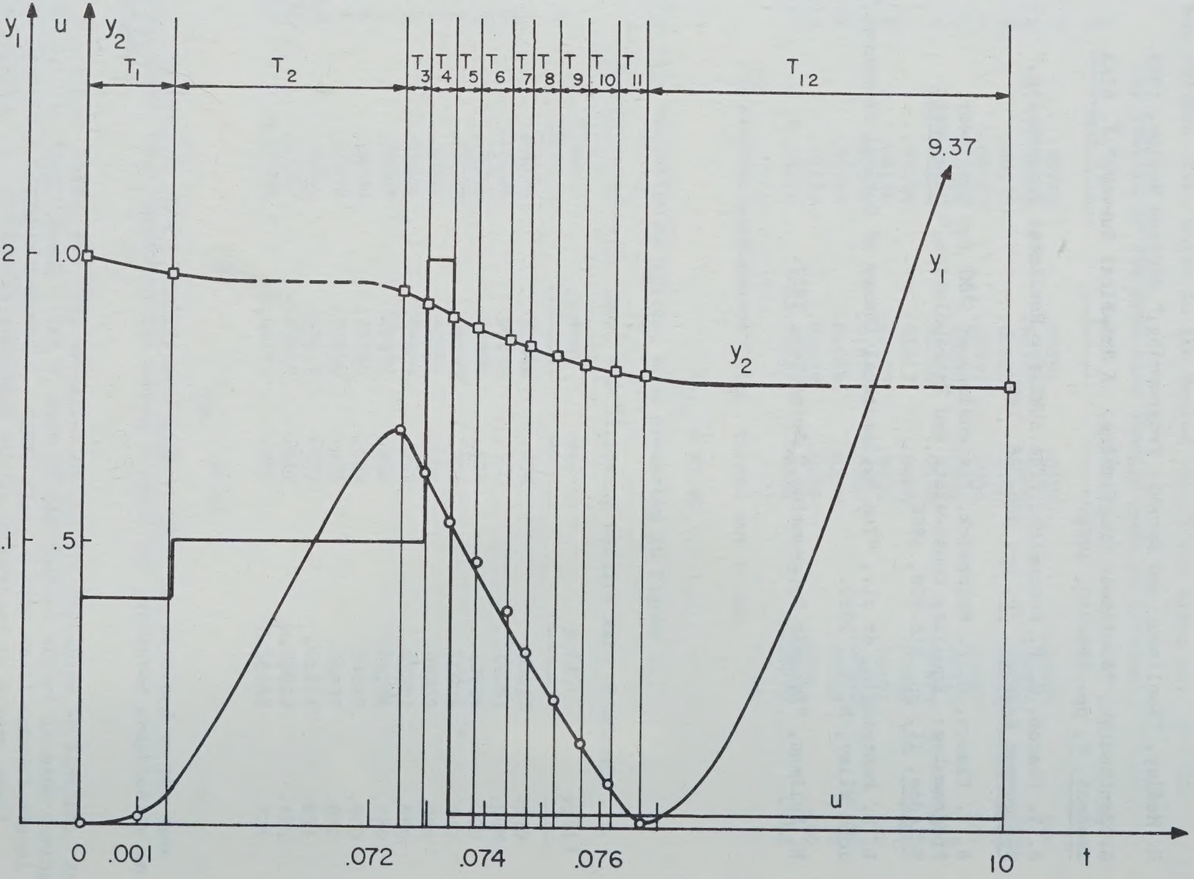

y1
u
y2
T1
T2
T3
T4
T5
T6
T7
T8
T9
T10
T11
T12
9.37
.2
1.0
.1
.5
y1
y2
u
0
.001
.072
.074
.076
10
t

Figure 1

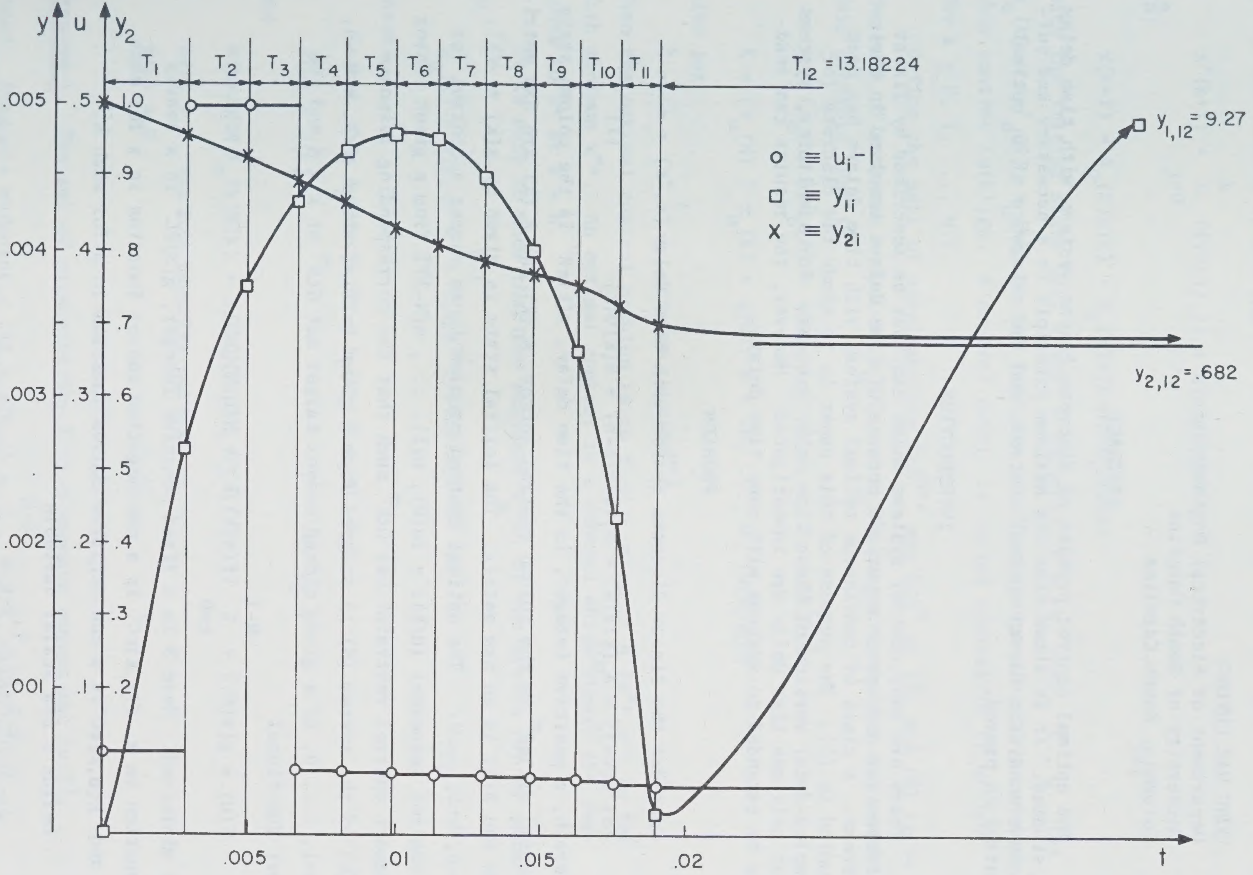

y
u
y_2
T_1
T_2
T_3
T_4
T_5
T_6
T_7
T_8
T_9
T_{10}
T_{11}
T_{12} = 13.18224
$y_{1,12}$ = 9.27
$y_{2,12}$ = .682
o ≡ $u_i - 1$
□ ≡ y_{1i}
x ≡ y_{2i}
.005
.004
.003
.002
.001
.5
.4
.3
.2
.1
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.005
.01
.015
.02
t

Figure 2

DISCRETE SYSTEMS WITH TIME DELAY

DONG HAK CHYUNG
Department of Electrical Engineering
University of South Carolina
Columbia, South Carolina

ABSTRACT

The optimal control problem of discrete linear systems with time delay is studied. It is shown that the maximum principle is a necessary and sufficient condition for an optimal control, and the existence of an optimal control is proved.

INTRODUCTION

There are many control systems which can best be described by linear differential-difference equations because of time delays involved in these systems. A class of continuous optimal systems with time delays has been studied in [1]. The purpose of this paper is to study the discrete (or sampled-data) version of these time delay systems. For simplicity, systems with only one time delay are investigated. However, the results can readily be extended to systems with many time delays.

PROBLEM

Consider the linear discrete system with time delay

$$(S)\quad x(k+1) = A_0(k)x(k) + A_1(k)x(k-h) + B(k)u(k) \qquad (1)$$

where h, a positive integer, is the time delay, $x(k)\varepsilon R^n$ is the column state vector, $u(k)\varepsilon R^m$ is the column control vector, $A_0(k)$ and $A_1(k)$ are nxn matrices and B(k) is an nxm matrix. The initial state is given by $x(k) = \phi(k)$, $k=-h,-h+1,\ldots,0$. The optimal control problem is to choose a control (or a control sequence) $\{u(k)\} = \{u(0), u(1), \ldots, u(N-1)\}$ from a given convex compact control restraint set $\Omega\subset R^m$ such that the corresponding response x(k) of the system (S) is steered from a given initial state $\phi(k)$, $k = -h, -h+1, \ldots, 0$, to a given closed convex target set $G\subset R^n$ at k = N and the cost functional

$$C(u) = g(x(N)) + \sum_{k=0}^{N-1} \{f(x(k),k) + h(u(k),k)\} \qquad (2)$$

is minimized. Here N is a fixed positive integer, $g(x)\varepsilon C^1$ is a convex function in x, $f(x,k)\varepsilon C^1$ is a non-negative convex function in x for each k and $h(u,k)\varepsilon C^\circ$ is a non-negative convex function in u for each k.

Define a new scalar variable

$$x^\circ(k) = \sum_{j=0}^{k-1} \{f(x(j),j) + h(u(j),j)\},\ x^\circ(0) = 0 \qquad (3)$$

and let $\hat{x}(k) = (x^\circ(k), x(k)) \varepsilon R^{n+1}$ be a column vector. Then define a new (n+1)-dimentional system $(\hat{S})$

$$(\hat{S}) \quad \begin{aligned} x^\circ(k+1) &= \sum_{j=0}^{k} \{f(x(j),j) + h(u(j),j)\} \\ x(k+1) &= A_0(k)x(k) + A_1(k)x(k-h) + B(k)u(k) \end{aligned} \tag{4}$$

with the initial state $\hat{x}(k) = \hat{\phi}(k) = (0, \phi(k))$, $k = -h, -h+1, \ldots, 0$. Let $\hat{x}_u(k) = (x^\circ_u(k), x_u(k))$ be the response of $(\hat{S})$ corresponding to an admissible control $\{u(k)\} \subset \Omega$. A control $\{u(k)\}$ is called admissible if $u(k) \varepsilon \Omega$ for $k = 0, 1, \ldots, N-1$.

Define the sets of attainability $\hat{K} \subset R^{n+1}$ and $K \subset R^n$ of $(\hat{S})$ and (S) to be the sets of response endpoints $\hat{x}_u(N)$ and $x_u(N)$ respectively for all admissible controls, that is,

$$\hat{K} = \{\hat{x}_u(N) \mid \hat{x}_u(k) = \hat{\phi}(k),\ k = -h, \ldots, 0,\ \{u(k)\} \subset \Omega\}$$

$$K = \{x_u(N) \mid x_u(k) = \phi(k),\ k = -h, \ldots, 0,\ \{u(k)\} \subset \Omega\}$$

Also let

$$\hat{G} = \{\hat{x} = (x^\circ, x) \mid -\infty < x^\circ < \infty,\ x \varepsilon G\} \subset R^{n+1}$$

Then the optimal control problem is to find a point $\hat{x}^* = (x^{\circ *}, x^*)$ in $\hat{K} \cap \hat{G}$ with minimum $x^{\circ *}$. An optimal control is a control which steers the response to $\hat{x}^*$ at $k = N$.

It is easy to show that the response $x_u(k)$ of the system (S) is given by

$$x_u(k) = x(k,\phi) + \sum_{j=0}^{k-1} W(k)W^{-1}(j)B(j)u(j-1)\ ) \tag{5}$$

where

$$x(k+1,\phi) = A_o(k)x(k,\phi) + A_1(k)x(k-h,\phi),\ k \geq 0$$

$$x(k) = \phi(k) \qquad ,\ -h \leq k \leq 0$$

and

$$W(k+1) = A_o(k)W(k) + A_1(k)W(k-h),\ k \geq 0$$

$W(0) = I$, the nxn identity matrix

$$W(k) = 0 \qquad ,\ k < 0$$

SET OF ATTAINABILITY

Theorem 1. The set of attainability K is a compact convex set in R^n.

Proof. Response endpoint $x_u(N) = x(N,\phi) + \sum_{k=0}^{N-1} W(N)W^{-1}(k)B(k-1)u(k-1)$

is a linear continuous function of $u(0)$, $u(1)$, ..., $u(N-1)$. Therefore $K = \{x_u(N) \mid \{u(k)\} \subset \Omega\}$ is a linear mapping of $\Omega \times \Omega \times \ldots \times \Omega \subset R^{Nm}$ into R^n. Furthermore $\Omega \times \Omega \times \ldots \times \Omega$ is a compact convex set in R^{Nm}, for Ω is a compact convex set in R^m. Therefore K is a compact convex set in R^n.

Theorem 2. The set of attainability $\hat{K}$ is a compact set in R^{n+1}.

Proof. $\hat{x}_u(N) = (x^\circ_u(N), x_u(N))$ is a continuous function of $u(0)$, $u(1)$, ..., $u(N-1)$, for $x^\circ_u(N)$ is continuous in $x_u(k)$ and $u(k)$, $k = 0, 1, \ldots, N-1$, and each $x_u(k)$ is continuous in $u(0)$, $u(1)$, .., $u(k-1)$. Therefore $\hat{K} = \{\hat{x}_u(N) \mid \{u(k)\} \subset \Omega\}$ is again a continuous mapping of the compact set $\Omega \times \Omega \times .. \times \Omega \subset R^{Nm}$ into R^n. Thus $\hat{K}$ is compact in R^{n+1}.

Assume that $\hat{K}$ has a nonempty interior in R^{n+1}. Otherwise the same analysis can be done in the linear manifold which is spanned by $\hat{K}$ in R^{n+1}. Since $x^\circ_u(N) \geq 0$ for all $\{u(k)\} \subset \Omega$, $\hat{K}$ is in the upper halfspace (with respect to x°-axis) $x^\circ \geq 0$ of R^{n+1}. Since $\hat{K}$ is compact, define the lower boundary $\partial\hat{K}$ of $\hat{K}$ to be the set of all lower limit points $\hat{x} = (x^\circ, x)$ in $\hat{K}$ such that $x^\circ = \min \{y^\circ\}$ for all (y°, x) in $\hat{K}$ for each fixed x, that is

$$\partial\hat{K} = \{\hat{x} \mid \hat{x} = (x^\circ, x) \varepsilon K,\ x^\circ = \min \{y^\circ\} \text{ for all } (y^\circ, x) \varepsilon \hat{K}\}.$$

From the definitions of $\hat{K}$ and K it is apparent that K is an orthogonal projection of $\hat{K}$ on the x-space R^n given by $x^\circ = 0$ in R^{n+1}, and so $\partial\hat{K}$ is a hypersurface over K in (x°, x)-space R^{n+1}.

Theorem 3. The lower boundary $\partial\hat{K}$ is a convex hypersurface over the compact set K in R^{n+1}.

Proof. Consider any two points $\hat{x}_1(N)$ and $\hat{x}_2(N)$ on $\partial\hat{K}$. Let $\{u_1(k)\}$, and $\{u_2(k)\}$ be the corresponding controls. If there exists a point $\hat{x}$ in $\hat{K}$ such that $x = \lambda x_1(N) + (1-\lambda) x_2(N)$ and $x^\circ \leq \lambda x^\circ 1(N) + (1-\lambda) x^\circ_2(N)$ for all $0 \leq \lambda \leq 1$ then $\partial\hat{K}$ is a convex hypersurface over K, for then there exists a point (y°, y) on $\partial\hat{K}$ with $y = x$, $y^\circ \leq x^\circ$.

Define a new control $\{u_\lambda(k)\}$ by $u_\lambda(k) = \lambda u_1(k) + (1-\lambda) u_2(k)$. Obviously $\{u_\lambda(k)\} \subset \Omega$ for Ω is convex. Let $\hat{x}_\lambda(k)$ be the corresponding response. Then $\hat{x}_\lambda(N) \varepsilon \hat{K}$. From the quation (5) and the convexity of f and h, it is easy to show (see [2],[4]) that $x_\lambda(N) = \lambda x_1(N) + (1-\lambda) x_2(N)$ and $x^\circ_\lambda(N) \leq \lambda x^\circ_1(N) + (1-\lambda) x^\circ_2(N)$. Therefore $\partial\hat{K}$ is a convex hypersurface.

OPTIMAL CONTROL

It is later shown that an optimal point $\hat{x}^*$ in $\hat{K}$ is on the lower boundary $\partial\hat{K}$. The following maximum principle gives necessary and sufficient conditions for a control to steer the response to $\partial\hat{K}$.

Theorem 4. A control $\{u^*(k)\} \subset \Omega$ steers its response $\hat{x}^*(k)$ to the lower boundary $\partial\hat{K}$ of $\hat{K}$ if and only if there exists a solution $\hat{q}(k) = (q_o, q(k)) \varepsilon R^{n+1}$ of the adjoint equation

$q_o \le 0$; constant

$$q(k) = \begin{cases} q(k+1)A_o(k) + q_o \frac{\partial f}{\partial x}\,(x^*(k),k), & N-h \le k < N \\ q(k+1)A_o(k) + q(k+1+h)A_1(k+h) + q_o \frac{\partial f}{\partial x}\,(x^*(k),k), & 0 \le k < N-h \end{cases} \tag{6}$$

such that

$$q_o h(u^*(k),k) + q(k+1)B(k)u^*(k) = \max_{u\in\Omega} q_o h(u,k) + q(k+1)B(k)u$$

for all k.

Proof. Suppose $\{u^*(k)\}\subset\Omega$ with $\hat{x}^*(k), \hat{q}(k)$ satisfies the maximum principle. Let $\{u(k)\}\subset\Omega$ be an arbitrary control with response $\hat{x}(k)$. Then

$$\hat{q}(N)\hat{x}(N) = \sum_{k=0}^{N-1} \{\hat{q}(k+1)\hat{x}(k+1) - \hat{q}(k)\hat{x}(k)\} + \hat{q}(0)\hat{x}(0)$$

$$= \sum_{k=0}^{N-1} \{q_o x^\circ(k+1) + q(k+1)x(k+1) - q_o x^\circ(k) - q(k)x(k)\} + q(0)\phi(0)$$

Substitute the equations (4), (5), (6) for $x^\circ(k+1)$, $x^\circ(k)$, $x(k+1)$, $q(k)$. Then

$$\hat{q}(N)\hat{x}(N) = \sum_{k=0}^{N-1} q_o\{f(x(k),k) - \frac{\partial f}{\partial x}\,(x^*(k),k)x(k)\}$$

$$+ \sum_{k=0}^{N-1} \{q(k+1)B(k)u(k) + q_o h(u(k),k)\}$$

$$+ \sum_{k=0}^{h-1} q(k)A_1(k)\phi(k-h) + q(0)\phi(0)$$

For $\{u^*(k)\}$,

$$\hat{q}(N)\hat{x}^*(N) = \sum_{k=0}^{N-1} q_o\{f(x^*(k),k) - \frac{\partial f}{\partial x}\,(x^*(k),k)x^*(k)\}$$

$$+ \sum_{k=0}^{N-1} \{q(k+1)B(k)u^*(k) + q_o h(u^*(k),k)\}$$

$$+ \sum_{k=0}^{h-1} q(k)A_1(k)\phi(k-h) + q(0)\phi(0)$$

By the maximum principle

$$q(k+1)B(k)u^*(k) + q_o h(u^*(k),k) \ge q(k+1)B(k)u(k) + q_o h(u(k),k)$$

and from the convexity of $f(x,k)$ and $q_o \le 0$

$$q_o\{f(x^*(k),k) - \frac{\partial f}{\partial x}\,(x^*(k),k)x^*(k)\} \ge q_o\{f(x(k),k) - \frac{\partial f}{\partial x}\,(x^*(k),k)x(k)\}$$

Therefore by comparing the above two equations

$$\hat{q}(N)\hat{x}^*(N) \ge \hat{q}(N)\hat{x}(N)$$

for any $\hat{x}(N)\varepsilon\hat{K}$. Thus $\hat{x}^*(N)$ is a boundary point of the compact set $\hat{K}$. If $q_o<0$ then obviously $\hat{x}^*(N)$ is a lower boundary point. If $q_o = 0$ then $x^*(N)$ is a boundary point of K and so $\hat{x}^*(N)$ is a lower boundary point of $\hat{K}$ (See [2]). Hence $\hat{x}^*(N)\varepsilon\partial K$.

Conversely, suppose $\{u^*(k)\}\subset\Omega$ steers the response $\hat{x}^*(k)$ to $\partial\hat{K}$ at $k = N$. Then since $\partial\hat{K}$ is a convex hypersurface, there exists an exterior normal vector $\hat{p} = (p^\circ,p)$ to $\hat{K}$ at $\hat{x}^*(N)$. Obviously $p^\circ\leq 0$. Let $\hat{q}(k) = (q_o,q(k))$ be the solution of the adjoint equation (6) with $\hat{q}(N) = \hat{p}$. If $\{u^*(k)\}$ does not satisfy the maximum principle with respect to the adjoint response $\hat{q}(k)$ at some k, say k_o, then for any small $\varepsilon>0$, there exists a $u\varepsilon\Omega$ such that

$$q_oh(u,k_o) + q(k_o+1)B(k_o)u = q_oh(u^*(k_o) + q(k_o+1)B(k_o)U^*(k_o) + \varepsilon$$

$$\leq \max_{u\varepsilon\Omega} q_oh(u,k_o) + q(k_o+1)B(k_o)u$$

Define a new control $\{u_\varepsilon(k)\}\subset\Omega$ by $u_\varepsilon(k) = u(k)$, $k \neq k_o$ and $u_\varepsilon(k_o) = u$. Let $\hat{x}_\varepsilon(k)$ be the corresponding response. Then a similar calculation as before shows (See [3]) that

$$\hat{q}(N)\hat{x}^*(N) -\hat{q}(N)\hat{x}_\varepsilon(N) <c\varepsilon^2 - \varepsilon$$

for some constant c. Therefore, for sufficiently small $\varepsilon >0$

$$\hat{q}(N)\hat{x}^*(N)< \hat{q}(N)\hat{x}_\varepsilon(N)$$

But this is impossible for $\hat{q}(N)$ is an exterior normal vector to $\hat{K}$ at $\hat{x}^*\varepsilon\partial\hat{K}$ with $q_o \leq 0$, the lower boundary $\partial\hat{K}$ is a convex hypersurface and $\hat{x}_\varepsilon(N)\varepsilon\hat{K}$. Therefore $\{u^*(k)\}$ satisfies the maximum principle.

Theorem 5. Suppose $G = R^n$ (free endpoint problem). Then there exists an optimal control. Furthermore there exists a solution $x^*(k),q^*(k)$ of

$$x(k+1) + A_o(k)x(k) + A_1(k)x(k-h) + B(k)u^*(k,q)$$

$$q(k) =\begin{cases} q(k+1)A_o(k) + q_o \frac{\partial f}{\partial x}(x(k),k) & N-h\leq k< N \\ q(k+1)A_o(k) + q(k+1+h)A_1(k+h) + q_o\frac{\partial f}{\partial x}(x(k),k), & 0\leq k< N - h \end{cases}$$

with the initial contition $x(k) = \phi(k)$, $k = -h, -h+1, ---, 0$ and the transversality condition $q(N) = -\operatorname{grad} q(x(N))$, $q_o = -1$. Here $u^*(k,q)$ is defined by the maximum principle

$$q(k+1)B(k)u^*(k,q) + q_oh(u^*(k,q),k) = \max_{u\varepsilon\Omega} q(k+1)B(k)u + q_oh(k,u)$$

$u^*(k) = u^*(k,q^*)$ is an optimal control.

Proof. The cost functional $C(u) = g(x(N)) + x^\circ(N)$ is continuous in $\hat{x}(N)$ and $\hat{K}$ is compact. Therefore there exists an optimal point $\hat{x}^*(N)$, a point where $C(u)$ is minimum, in $\hat{K}$. Furthermore, since $C(u)$ is strictly increasing in $x^\circ(N)$, $\hat{x}^*(N)$ must be on the lower boundary, Therefore, by Theorem

4, there exists an optimal control which satisfies the maximum principle.

On the other hand, if a control $\{u^*(k)\}$ satisfies the maximum principle with $q_0 = -1 < 0$, then by Theorem 4 the corresponding response end-point $\hat{x}^*(N)$ is on the lower boundary. Furthermore, the condition $q(N) = -\operatorname{grad} g(x^*(N))$ assures that $\hat{x}^*(N)$ is an optimal point. For more details see [2].

CONCLUSION AND REMARKS

A discrete linear optimal system with time delay has been studied. It has been shown that the maximum principle with appropriate transversality condition is a necessary and sufficient condition for an optimal control, and a method of finding an optimal control has been given in the form of two point boundary value problem.

Theorem 5 can be easily modified to handle the case when G is a closed convex subset of R^n (See [2]). Also a similar results can be obtained in the case when $\Omega = R^m$.

References.

1. Chyung, D.H., and Lee, E.B., "Linear Optimal Systems with Time Delays" SIAM J. on Control, Vol. 4, No. 3, November 1966, Pp. 548 - 575.
2. Chyung, D.H., "Discrete Linear Optimal Control System with Essentially Quadratic Cost Functional" IEEE Trans. on Automatic Control, Vol. AC-11. No. 3, July 1966, Pp. 404 - 413.
3. Chyung, D.H., "An Approximation to Bounded Phase Coordinate Control Problem for Linear Discrete Systems", IEEE Trans. on Automatic Control. Vol. AC-12, No.1, Feb. 1967, Pp 37 -42.
4. Lee, E.B., and Markus, L. "Foundations of Optimal Control Theory" John Wiley, New York, 1967.
5. Pontryagin, L.E. et al., "The Mathematical Theory of Optimal Processes", Interscience, New York, 1962.

A METHOD FOR EVALUATING MULTIPLE MODES OF LINEAR SYSTEMS

C. F. CHEN
E. E. Dept.
University of Houston
Houston, Texas

D.W.C. SHEN
Moore School of E. E.
University of Penn.
Philadelphia, Pa.

ABSTRACT

The Newton method of successive approximations for finding multiple modes, real or complex, of a large linear system presents great difficulty because of the slow convergence property in the neighborhood of a multiple mode. This paper describes a gradient method with rapid convergence property which reduces considerably the amount of computation involved.

The gradient method arrives at a modified Newton equation as follows.

$$S_2 = S_1 - m \frac{W}{W'}$$

where m is the multiplicity of the repeated root and W = f(s). If m is not known, starting from m = 1, we can increase m by unity each time until the multiple root is found. Examples are given to demonstrate the feasibility and the simplicity of this method.

OSCILLATIONS IN SYSTEMS WITH EVEN NONLINEARITIES

SAMUEL A. MUSA
The Moore School of Electrical Engineering
University of Pennsylvania
Philadelphia, Pennsylvania

Subharmonic and superharmonic oscillations in weakly nonlinear systems with sinusoidal excitation and restricted classes of nonlinearities are investigated. The problem under consideration is that of a system excited at an angular frequency ω, and whose response consists of one principal component at frequency ω and another at frequency $\frac{n}{m}\omega$ where n and m are integers ($\frac{n}{m} < 1$ denotes subharmonics and $\frac{n}{m} > 1$ represents superharmonics). The system is described by the differential equation:

$$\ddot{x} + \epsilon f(x,\dot{x}) + \omega_o^2 x = F \cos \omega t$$

where $0 < \epsilon \ll 1$, f represents the nonlinearities, F is of order unity in magnitude, and the detuning $(\frac{n^2}{m^2}\omega^2 - \omega_o^2)$ is of order ϵ.

For nonlinearities even in x and $\dot{x}$, the orders m and n cannot be both odd integers or both even integers with an odd ratio. Furthermore, based on the order of approximation which led to the prediction of the possible orders $\frac{n}{m}$, all sub- and superharmonic oscillations other than the trivial zero solutions in systems with even strictly monotonic and continuous nonlinearities are shown to be unstable. The instability of the amplitude-phase configuration of the subharmonics and superharmonics is described by saddle-type of singularities.

Oscillations with orders odd m and even n and vice versa cannot be sustained in systems with odd monotonely nondecreasing nonlinearities $f(\dot{x})$ or $f(x)$ and nonpositive detuning.

FINITE TIME STABILITY OF MULTI NON-LINEAR CONTROL SYSTEMS*

A. N. MICHEL and S. H. WU
Department of Electrical Engineering
Marquette University
Milwaukee, Wisconsin

In many cases of practical interest, there is concern over the behavior of dynamical systems only over a fixed, finite interval of time. For example, the booster phase of a rocket flight through the earth's atmosphere is of a finite duration, the lifetime of a chemical process is of finite time, etc. Thus, it appears to be of interest to investigate certain dynamical systems within a stability framework which is defined over a finite interval of time. This was precisely done by L. Weiss and E. F. Infante [1], [2].

In their two papers, Weiss and Infante [1], [2], first gave several definitions of "finite time stability" of dynamical systems, and then proceeded to establish sufficient conditions for the "finite time stability" of such systems. Weiss and Infante also considered the problem of "finite time stability" of systems which are defined on product spaces. Again several definitions of "finite time stability" of systems defined on product spaces are given and sufficient conditions for such stability are established.

In this paper some of the results given in [2] concerning the "finite time stability" of systems defined on product spaces are modified and extended. Utilizing these results, sufficient conditions for the "finite time stability" of the following types of large systems (i.e. systems which consist of many subsystems) are established:

a. n subsystems connected in cascade, where each subsystem is endowed with an "admissible" non-linearity, and where the entire system is connected in a feedback configuration.
b. n subsystems connected in parallel, where each subsystem is endowed with an "admissible" non-linearity, and where the entire system is connected in a feedback configuration.

* This work was supported in part by the National Science Foundation under Grant No. GK-855.

[1] Leonard Weiss and E. F. Infante, "On The Stability Of Systems Defined Over A Finite Time Interval", Proc. Nat'l Acad. Sci., Vol. 54, pp. 44-48, July 1965.

[2] Leonard Weiss and E. F. Infante, "Finite Time Stability Under Perturbing Forces And On Product Spaces", IEEE Trans. On Automatic Control, Vol. A.C.-12, pp. 54-59, February 1967.

A NEW METHOD FOR SOLVING PA + A^{T}P=-D

C. F. CHEN
E. E. Dept.
University of Houston
Houston, Texas

L. S. SHIEH
E. E. Dept.
University of Houston
Houston, Texas

ABSTRACT

Constructing Liapunov functions for a linear system

$$[\dot{x}] = [A]\ [x] \tag{1}$$

involves the calculation of the following matrix equation

$$[P]\ [A] + [A]^T [P] = -[D] \tag{2}$$

by expanding it into n(n+1)/2 equations. If n is large, to expand the matrix equation is rather complicated.

A new method is developed by converting (2) into the following equation,

$$[\alpha]\ \vec{p}_\nu = -\tfrac{1}{2}\vec{d}_\nu \tag{3}$$

where $\vec{p}_\nu$ and $\vec{d}_\nu$ are the vectors which are organized by the elements of [P] and [D] respectively. $[\alpha]$ can be found by several simple rules.

We use (K,L) and (I,J) as the row and column subscripted index respectively. The following restrictions should be satisfied when forming the $[\alpha]$ matrix directly from the [A] matrix.

(1) if $K = I,\ L \neq J,\ \longrightarrow A(L,J)$
$K \neq I,\ L = J,\ \longrightarrow A(K,I)$

(2) $K \neq I,\ L \neq J,$
$K = J,\ L \neq I,\ \longrightarrow A(L,I)$
$K \neq J,\ L = I,\ \longrightarrow A(K,J)$
$K \neq J,\ L \neq I,\ \longrightarrow 0$

(3) $K = I,\ L = J$
$K = J,\ L = I,\ \longrightarrow A(K,I)$
$K \neq J,\ L \neq I,\ \longrightarrow A(K,I) + A(L,J)$.

For a third order equation, $[\alpha]$ reads: (K,L)

(I,J)	1-1	1-2	1-3	2-2	2-3	3-3
1-1	A_{11}	A_{21}	A_{31}	0	0	0
1-2	A_{12}	$A_{11} + A_{22}$	A_{32}	A_{21}	A_{31}	0
1-3	A_{13}	A_{23}	$A_{11} + A_{33}$	0	A_{21}	A_{31}
2-2	0	A_{12}	0	A_{22}	A_{32}	0
2-3	0	A_{13}	A_{12}	A_{23}	$A_{22} + A_{33}$	A_{32}
3-3	0	0	A_{13}	0	A_{23}	A_{33}

where the subscripted index (K,L) and (I,J) are the comparing index. For example, we want to find the element Z.

$$\begin{array}{c|c} & (K\text{-}L) \\ \hline (I\text{-}J) & Z = ? \end{array}$$

We examine that (1) if $K = I = L = J$, we have $Z = A(K,I)$
(2) if $K \neq I,\ L \neq J$ and $K \neq J,\ L \neq I$, then $Z = 0$.

etc.

PERIODICALLY TIME-VARYING SYSTEM: AN INSTABILITY CRITERION

THEODORE A. BICKART
Electrical Engineering Department
Syracuse University
Syracuse, New York 13210

ABSTRACT

Two theorems on the instability of a periodically time-varying system are stated. Their relationship to recently published results on instability is discussed. Proofs, made elementary by the assumption of periodicity, are given.

RESULTS

Consider the differential equation

$$\sum_{i=0}^{n} [a_i + k(t)b_i] \frac{d^i x}{dt^i} = 0, \qquad (1)$$

under the following assumptions: (A1) The a_i and b_i, $i = 0, 1, \ldots, n$, are real constants with $a_n \neq 0$, (A2) t is a real variable, (A3) $k(t)$ is a real-valued, piece-wise continuous, periodic function — $k(t) = k(t + T)$— of period T, where T is a real, positive constant, and (A4) $[a_n + k(t)b_n] > 0$ for $t \in [0,T]$.

Let $\mathfrak{D} \subset \mathfrak{R}$, $\mathfrak{R}$ denotes the real line, be the set of points at which $k(t)$ is discontinuous. Any complex-valued function which is n-1 times continuously differentiable on $\mathfrak{R}$ and satisfies (1) on $\mathfrak{R} - \mathfrak{D}$ is called a solution of (1).

It is convenient to introduce the following definitions: (D1) k_o is any constant such that $[a_n + k_o b_n] > 0$, (D2) $\hat{k}(t) = k(t) - k_o$, (D3) $K = \mathrm{SUP}_{t\in[\mathfrak{R}-\mathfrak{D}]} |\hat{k}(t)|$, and (D4) $H(\omega) = \Sigma_{i=0}^{n} b_i(j\omega)^i / \Sigma_{i=0}^{n}[a_i + k_o b_i](j\omega)^i$, with $j = \sqrt{-1}$.

The principle results being reported are the proofs of the following two theorems.

Theorem 1 Let (A1) through (A4) be valid. Further, if there exists a k_o such that (H1) $\Sigma_{i=0}^{n}[a_i + k_o b_i]s^i$ has no zeros with zero real part and one or more zeros with positive real part and (H2) $K\ \mathrm{SUP}_{\omega\in\mathfrak{R}}|H(\omega)| \leq 1$, then at least one solution of (1) does not approach zero as $t \to \infty$.

Theorem 2 Let (A1) through (A4) be valid. Further, if there exists a k_o such that (H1) $\Sigma_{i=0}^{n}[a_i + k_o b_i]s^i$ has no zeros with zero real part and one or more zeros with positive real part and (H3) $K\ \mathrm{SUP}_{\omega\in\mathfrak{R}}|H(\omega)| < 1$, then at least one solution of (1) is not bounded as $t \to \infty$.[1]

---

[1] (H3) can be replaced by a less restrictive hypothesis at the expense of a slightly lengthened proof. The proof is to be found in [1].

DISCUSSION

Suppose $\alpha \leq k(t) \leq \beta$, where α and β are distinct, real constants. Then, let $k_0 = (\alpha + \beta)/2$ and get, as a consequence, $K = (\beta - \alpha)/2$. Further, let $G(s) = \Sigma_{i=0}^{n} b_i s^i / \Sigma_{i=0}^{n} a_i s^i$ and z be the number of zeros of G(s) with positive real part. The interpretation of (H1) and (H2) is: The inverse Nyquist diagram of G(s) must not intersect the open disk, centered at $(-(\alpha+\beta)/2, 0)$ in the complex plane, of radius $(\beta-\alpha)/2$ and must encircle it (1-z) or more times in the clockwise direction. The interpretation of (H1) and (H3) is similar; "open disk" is replaced by "closed disk" in the preceding statement. Now, suppose α and β are of the same sign and let p be the number of poles of G(s) with positive real part. Then, an alternate interpretation of (H1) and (H2) is: The Nyquist diagram of G(s) must not intersect the open disk, centered at $(-(1/\alpha + 1/\beta)/2, 0)$ in the complex plane, of radius $|(1/\alpha - 1/\beta)/2|$ and must encircle it (1-p) or more times in the clockwise direction. The corresponding alternate interpretation of (H1) and (H3) replaces "open disk" with "closed disk". Under the added restrictions that produce these latter interpretations, the conditions stated in the theorems differ in only one significant respect from those for the first half of each of the first two instability theorems recently given by Brockett and Lee [2]. In their underlying assumptions, Brockett and Lee did not invoke a periodicity condition on k(t); their theorems are, therefore, applicable to a larger class of differential equations. Because of these facts, the primary object of this note is to present proofs which are significantly simpler by virtue of the periodicity assumption (A3).[2]

PROOFS

The proofs which follow were derived, with only minor changes, from the proof given by Sandberg [3] for the asymptotic stability of (1).

Proof 1 Consider

$$\sum_{i=0}^{n} [a_i + k_0 b_i + \delta \hat{k}(t) b_i] \frac{d^i x}{dt^i} = 0 \tag{2}$$

for $\delta \epsilon [0,1]$. By (A4), (D1), and (D2), $[a_n + k_0 b_n + \delta \hat{k}(t) b_n] > 0$ for all $(t,\delta) \in [0,T] \times [0,1]$. Therefore, for any $\delta \in [0,1]$ the solution space of (2) is spanned by a set of solutions of the Floquet form [4]. Further, it is known [5] that the characteristic exponents are continuous functions of δ in $[0,1]$.

---

[2] Brockett and Lee established instability in terms of the Euclidian norm of the state vector not approaching zero or not being bounded as $t \to \infty$ for some initial state vector; since the theorems in this note establish similar behavior for x — an element of their state vector — conclusions of the theorems presented here are not viewed as materially different from theirs.

The proof will be by contradiction. Thus, suppose all solutions of (2) for $\delta=1$ approach zero as $t \to \infty$. Then, for $\delta=1$ all of the characteristic exponents must have negative real part. By (H1), one or more of the characteristic exponents have positive real part and none have zero real part for $\delta=0$. Therefore, a $\delta_o \in (0,1)$ must exist such that (2) possesses a solution of the form $y(t)e^{j\nu t}$, where ν is a real constant, $y(t)$ is a complex-valued function of period T, and $y(t) \not\equiv 0$ on $[0,T]$. Observe: Without loss of generality, it may be assumed that $|\nu| \leq \pi/T$. Note: The proof by contradiction will be accomplished by showing that $y(t) \equiv 0$ on $[0,T]$.

Set $y(t) = \Sigma_{m=-\infty}^{+\infty} y_m e^{jm\omega_o t}$, where $\omega_o = 2\pi/T$ and $y_m = \frac{1}{T}\int_0^T y(t)e^{-jm\omega_o t}dt$. Then, since $y(t)e^{j\nu t}$ is a solution of (2) for $\delta = \delta_o$,

$$x(t) = \sum_{m=-\infty}^{+\infty} y_m e^{j(m\omega_o+\nu)t} \tag{3}$$

and

$$\frac{d^i x(t)}{dt^i} = \sum_{m=-\infty}^{+\infty} y_m[j(m\omega_o + \nu)]^i e^{j(m\omega_o+\nu)t} \tag{4}$$

for $i = 1, \ldots, n$. Note: The differentiation term-by-term is valid, since $x(t) = y(t)e^{j\nu t}$ is a solution of (2) and any solution is n-1 times continuously differentiable on $\mathcal{R}$.

Put $A(\omega) = \Sigma_{i=0}^{n}[a_i + k_o b_i](j\omega)^i$, $B(\omega) = \Sigma_{i=0}^{n} b_i(j\omega)^i$, $u(t) = \Sigma_{i=0}^{n}[a_i + k_o b_i]d^i x/dt^i$, and $v(t) = \Sigma_{i=0}^{n} b_i d^i x/dt^i$. By (2)

$$u + \delta_o \hat{k}(t)v = 0 \tag{5}$$

Also, by (3) and (4), $u(t) = \Sigma_{i=-\infty}^{+\infty} A(m\omega_o + \nu)y_m e^{j(m\omega_o+\nu)t}$ and $v(t) = \Sigma_{i=-\infty}^{+\infty} B(m\omega_o + \nu)y_m e^{j(m\omega_o+\nu)t}$. Let

$$u_m = \frac{1}{T}\int_0^T u(t)e^{-j(m\omega_o+\nu)t}dt = A(m\omega_o + \nu)y_m \tag{6}$$

and

$$v_m = \frac{1}{T}\int_0^T v(t)e^{-j(m\omega_o+\nu)t}dt = B(m\omega_o + \nu)y_m \tag{7}$$

Then, $v_m = [B(m\omega_o + \nu)/ A(m\omega_o + \nu)]u_m$, which by (D4) becomes $v_m = H(m\omega_o + \nu)u_m$. Consequently, the identity $\frac{1}{T}\int_0^T|v(t)|^2dt = \Sigma_{m=-\infty}^{+\infty}|v_m|^2$ yields

$$\frac{1}{T}\int_0^T |v(t)|^2 dt \leq \left\{ \underset{-\infty \leq m \leq +\infty}{\mathrm{SUP}} |H(m\omega_o + \nu)|^2 \right\} \sum_{m=-\infty}^{+\infty} |u_m|^2. \qquad (8)$$

Now, by (5) and (D3), the identity $\Sigma_{m=-\infty}^{+\infty} |u_m|^2 = \frac{1}{T}\int_0^T |u(t)|^2 dt$ yields

$$\sum_{m=-\infty}^{+\infty} |u_m|^2 \leq \delta_o^2 K^2 \frac{1}{T}\int_0^T |v(t)|^2 dt. \qquad (9)$$

Substitution of (9) in (8) establishes the inequality

$$\frac{1}{T}\int_0^T |v(t)|^2 dt \leq \delta_0^2 \{K \underset{-\infty \leq m \leq +\infty}{\mathrm{SUP}} |H(m\omega_o + \nu)|\}^2 \frac{1}{T}\int_0^T |v(t)|^2 dt \qquad (10)$$

Taking the supremum of the right hand side of (10) with respect to $|\nu| \leq \pi/T$ yields

$$\int_0^T |v(t)|^2 dt \leq \delta_0^2 \{K \underset{\omega \in \mathcal{R}}{\mathrm{SUP}} |H(\omega)|\}^2 \int_0^T |v(t)|^2 dt, \qquad (11)$$

since $\mathrm{SUP}_{|\nu| \leq \pi/T} \{\mathrm{SUP}_{-\infty \leq m \leq +\infty} |H(m\omega_o + \nu)|\} = \mathrm{SUP}_{\omega \in \mathcal{R}} |H(\omega)|$.

Now, by (H2) and the fact that $\delta_o < 1$, $\delta_o\{K\ \mathrm{SUP}_{\omega\in\mathcal{R}}|H(\omega)|\} < 1$ and, hence, $\int_0^T |v(t)|^2 dt = 0$. It follows from (5) that $\int_0^T |u(t)|^2 dt = 0$. Consequently, $u_m = 0$ for all m. Since $A(\omega) \neq 0$ for all real ω by (H1), it is clear, from (6), that $y_m = 0$ for all m. Since $y(t)e^{j\nu t}$ must have the properties of a solution of (2) for $\delta = \delta_o$, $y(t) \equiv 0$ on $[0,T]$ for all $\delta_o \in (0,1)$ and $|\nu| \leq \pi/T$.

Proof 2 The proof of Theorem 2 is obtained by replacing two paragraphs in the proof of Theorem 1

The second paragraph is replaced by: The proof will be by contradiction. Thus, suppose all solutions of (2) for δ=1 are bounded as $t \to \infty$. Then for δ=1 all of the characteristic exponents must have non-positive real part. By (H1), one or more of the characteristic exponents have positive real part and none have zero real part for δ=0. Therefore, a $\delta_o \in (0,1]$ must exist such that (2) possesses a solution of the form $y(t)e^{j\nu t}$, where ν is a real constant, $y(t)$ is a complex-valued function of period T, and $y(t) \not\equiv 0$ on $[0,T]$. Observe: Without loss of generality, it may be assumed that $|\nu| \leq \pi/T$. The proof by contradiction will be accomplished by showing that $y(t) \equiv 0$ on $[0,T]$.

The last paragraph is replaced by: Now, by (H3) and the fact that $\delta_o \leq 1$, $\delta_o\{K\ \mathrm{SUP}_{\omega\in\mathcal{R}}|H(\omega)|\} < 1$ and, hence, $\int_0^T |v(t)|^2 dt = 0$. It follows from (5) that $\int_0^T |u(t)|^2 dt = 0$. Consequently, $u_m = 0$ for all m. Since $A(\omega) \neq 0$ for all real ω by (H1), it is clear, from (6), that $y_m = 0$ for all m. Since $y(t)e^{j\nu t}$ must have the properties of a solution of (2) for $\delta = \delta_0$, $y(t) \equiv 0$ on $[0,T]$ for all $\delta_0 \in (0,1]$ and $|\nu| \leq \pi/T$.

FURTHER DISCUSSION

If the conditions of Theorem 2 are valid, then the number of characteristic exponents of (1) with positive real part can be determined easily. Recall: The proof of Theorem 2 focused on showing that no non-zero solution of (2) has an imaginary characteristic exponent for $\delta \in (0,1]$. Further, by (H1), no solution of (2) has an imaginary characteristic exponent for $\delta = 0$. Consequently, the number of characteristic exponents with positive real part is the same for all $\delta \in [0,1]$ and, in particular, for $\delta = 0$ and $\delta = 1$. Now, the number of characteristic exponents with positive real part for $\delta = 0$ is equal to the number of zeros of $\Sigma_{i=0}^{d}[a_i + k_0 b_i]s^i$ with positive real part. The latter is easily determined.

Let $n^- > -z$ be the number of times the inverse Nyquist diagram of G(s) encircles the point $(-k_0,0)$ in the complex plane in the clockwise direction. Then, $\Sigma_{i=0}^{d}[a_i + k_0 b_i]s^i$ has $n^- + z$ zeros with positive real part. Next, let $n^+ > -p$ be the number of times the Nyquist diagram of G(s) encircles the point $(-1/k_0,0)$ in the clockwise direction. Then, $\Sigma_{i=0}^{d}[a_i + k_0 b_i]s^i$ has $n^+ + p$ zeros with positive real part. Since the differential equation (1) is the same as (2) with $\delta = 1$, it may be concluded by the above comments that the number of characteristic exponents of (1) with positive real part is $n^- + z = n^+ + p$.

The next observation is: The hypotheses and conclusions of the Theorems are known, by the work of Brockett and Lee, to be valid when the periodicity assumption (A3) is not invoked. However, the proofs, which are substantially simpler under the periodicity assumption, are independent of the value of T. This appears to indicate that simpler proofs of the theorems given by Brockett and Lee may exist as variations on the proofs presented in this paper.

Finally, consider the system depicted in Figure 1. The signals e and r are related by the differential equation $\Sigma_{i=0}^{n} a_i d^i r/dt^i = \Sigma_{i=0}^{n} b_i d^i e/dt^i$ and the algebraic equation $e = s(t) - k(t)r$. The former equation may be replaced by a pair of equations $r = \Sigma_{i=0}^{n} b_i d^i x/dt^i$ and $e = \Sigma_{i=0}^{n} a_i d^i x/dt^i$. This pair of equations substituted in $e = s(t) - k(t)r$ yields $\Sigma_{i=0}^{n}[a_i + k(t)b_i]d^i x/dt^i = s(t)$, which for zero input is (1). With suitable assumptions it may be shown that the zero input response r(t) exhibits the same behavior as reported in the conclusions of the theorems for x(t). The second theorem in Sandberg's paper provides a set of suitable conditions. Alternately, the conditions by which Brockett and Lee establish the second half of each of their first two theorems are suitable; the conclusions and proof of those conclusions need only minor modification to adapt them to the manner in which the instability of (1) is presented by the theorems given in this note.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the discussions with Dr. I.W. Sandberg which led to the realization that the proof for the instability results presented here could be built upon his work on stability. Financial support for this research has been derived from the U.S. Air Force Rome Air Development Center contract AF 30(602)-3538 and the Syracuse University Research Corporation.

REFERENCES

1. T.A. Bickart, Instability of Periodically Time-Varying Linear Control Systems, SIAM Journal on Control, to be published.
2. R.W. Brockett and H.B. Lee, Frequency-Domain Instability Criteria for Time-Varying and Nonlinear Systems, Proc. IEEE, vol. 55, pp. 604-619, May 1967.
3. I.W. Sandberg, On the Stability of Solutions of Linear Differential Equations with Periodic Coefficients, SIAM Journal, vol. 12, pp. 487-496, June 1964.
4. P. Hartman, Ordinary Differential Equations, New York: John Wiley and Sons, 1964, pp. 60-62.
5. J.K. Hale, Oscillations in Nonlinear Systems, New York: McGraw-Hill, 1963, p. 21.

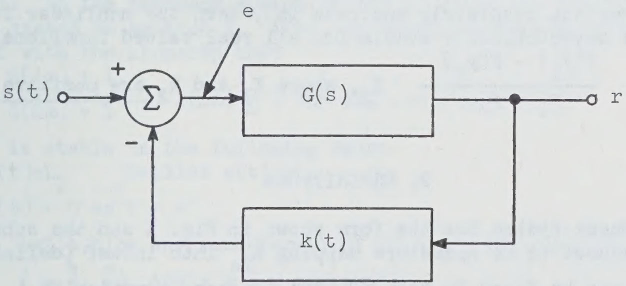


Figure 1 Single Loop Control System

AN OFF-AXIS CIRCLE CRITERION FOR THE STABILITY OF FEEDBACK SYSTEMS WITH A MONOTONIC NONLINEARITY

Yo Sung Cho
Honeywell Inc. Waltham, Mass.

Kumpati S. Narendra
Yale University, New Haven, Conn.

ABSTRACT: A circle criterion is stated in this paper as a sufficient condition for the stability of a nonlinear feedback system with a single monotonic nonlinearity. The criterion suggests a geometrical method for determining the sector

$$\left(K_1 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < K_2\right)$$

within which the nonlinearity should lie for asymptotic stability. The relation between the criterion stated and the well known Popov criterion and the circle criterion for time varying systems, is also discussed. Two examples are included to clarify various aspects of the criterion and its relation to existing results.

1. INTRODUCTION

Ever since Popov [1] stated the conditions for the stability of a nonlinear feedback system in the frequency domain, there has been considerable interest in deriving similar frequency domain conditions for systems with other types of nonlinearities. At the present time, the frequency conditions for determining the stability of systems with monotonic and odd monotonic nonlinearities are well established [2,3,4]. These conditions are expressed in terms of the transfer function $G(s)$ of the linear part of the system and a multiplier $Z(s)$ having a specific form, so that $G(s)\, Z(s)$ is positive real. However, most of the time it is not apparent from considering a transfer function how such a multiplier can be formed. In this paper, a geometrical condition is derived for the existence of an RL or an RC multiplier for the stability of feedback systems with monotonic nonlinearities. It is shown that if any circle can be drawn to intersect the negative real axis at $-\frac{1}{K_1}$ and $-\frac{1}{K_2}$ on the Nyquist plane so that the Nyquist plot $G(i\omega)$ lies outside the circle (and does not completely encircle it), then the nonlinear feedback system is asymptotically stable for all real valued functions $F(\cdot)$ satisfying $K_1 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < K_2$, where K_1 and K_2 are positive constants.

2. DEFINITIONS

If a feedback system has the form shown in Fig. 1 and the subsystems G and F are assumed to be operators mapping L_{2e} into intsef (definitions of L_2 and L_{2e} may be found in Ref. 2), the feedback equation in L_{2e} may be formulated as

$$\left.\begin{aligned} F\,G\,e(t) + e(t) &= x(t) \\ G\,e(t) &= y(t) \end{aligned}\right\} \qquad (2.1)$$

The objective is to find sufficient conditions under which $x(t) \varepsilon L_2$ and $e(t), y(t) \varepsilon L_{2e}$ in Eq. (2.1) implies $e(t), y(t) \varepsilon L_2$, and $y(\infty) = 0$.

2.1 Linear time-invariant operator G: G is assumed to satisfy the following conditions

$$Ge(t) = \int_0^t g(t - \tau)\, e(\tau) d\tau + g_1(t)$$

where the impulse response function g(t) is in $L \cap L_2$ and the Laplace transform of g(t) is

$$G(s) = \int_0^\infty g(t)\, e^{-st}\, dt$$

It is assumed that the function $g_1(t)$ resulting from initial conditions is in L_2.

2.2 Nonlinear function $F(\cdot)$: $F(\cdot)$ is a real valued function of a real variable and satisfies the following conditions:

(i) $F(y) = 0$ only if $y = 0$

(ii) $K_1 < \dfrac{F(y_1) - F(y_2)}{y_1 - y_2} < K_2$ for all real numbers y_1, y_2 satisfying $y_1 \neq y_2$.

K_1 and K_2 are real constants and satisfy the condition $-\infty < K_1 < K_2 < \infty$. Hence, if $K_1 \geq 0$, $F(\cdot)$ is a montonically increasing function.

3. PRELIMINARY RESULTS

Let RL be the class of complex valued functions Z(s) having the form:

$$Z(s) = \prod_{n=0}^{N} \frac{s + \alpha_n}{s + \beta_n}$$

where s is a complex variable and $0 < \alpha_i < \beta_i < \alpha_{i+1}$ (i=1, ...N) and let RC be the class of functions $Z^{-1}(s)$ for which Z(s) is in RL.

Theorem 3.1 and Lemma 3.1 stated in this section are used for deriving the circle criterion. Theorem 3.1 is a trivial extention of the well known multiplier theorem which may be found in corollary 3 of [2b], Theorem 3 of [3] and Theorem 4.1 of [4], and is stated without proof. Lemma 3.1 provides the specific form of the multiplier Z(s) in RL or RC which is needed for the proof of the geometrical criterion.

Theorem 3.1: In the feedback system of Fig.1, let G and F satisfy the conditions stated in section 2. Then, if there exists a function Z(s) in RL or RC with the property that

$$\mathrm{Re}\, \frac{K_2\, G(i\omega) + 1}{K_1\, G(i\omega) + 1}\, Z(i\omega) \geq 0 \text{ for all } -\infty < \omega < \infty,$$

the system is stable in the following sense

i) $x(t) \varepsilon L_2$ implies $y(t) \varepsilon L_2$

ii) $y(t) \to 0$ as $t \to \infty$

Lemma 3.1: Let [a,b] be any subinterval of $(0,\infty)$, θ any constant in $(-\frac{\pi}{2}, \frac{\pi}{2})$, and $\varepsilon > 0$ a constant.

i) if $\theta < 0$, then there is a function $Z(i\omega)$ in RL with the property that $|\mathrm{Arg.}\{Z(i\omega)\} - \theta| < \varepsilon$ for ω in [a,b].

ii) if $\theta < 0$, then there is a function $Z(i\omega)$ in RC with the property that $|\mathrm{Arg.}\ \{Z(i\omega)\} - \theta| < \varepsilon$ for ω in [a,b].

Proof of Lemma 3.1: Let $c = \dfrac{\theta}{\pi/2}$, $\alpha > 0$, $\rho > 1$ and n=1, ...N with the property that $\alpha < a$ and $\alpha \rho^{N+1} > b$.

Let $Z(i\omega) = \prod_{n=0}^{N} \frac{(i\omega + \alpha\rho^{n})}{(i\omega + \alpha\,\rho^{n+c})}$, then

$$\text{Arg.}\,\{Z(i\omega)\} = \sum_{n=0}^{N} \{\tan^{-1}\left(\frac{\omega}{\alpha\,\rho^{n}}\right) - \tan^{-1}\left(\frac{\omega}{\alpha\rho^{n+c}}\right)\}$$

Defining $f(\Omega) \equiv \tan^{-1}[\exp(\Omega)]$, $\Omega \equiv \log\omega$ and $\Omega_n = \log \alpha\rho^{n}$,

$$\text{Arg.}\,\{Z(i\omega)\} = \sum_{n=0}^{N} \{f(\Omega - \Omega_n) - f(\Omega - \Omega_{n+c})\}.$$

Defining $g(\Omega) = \sum_{n=0}^{N} u_{[\Omega_n,\, \Omega_{n+c}]}$ where $u_{[\Omega_n,\, \Omega_{n+c}]} = \begin{cases} 1 \text{ for } \Omega \text{ in } [\Omega_n, \Omega_{n+c}] \\ 0 \text{ elsewhere} \end{cases}$

$\text{Arg.}\,\{Z(i\omega)\} = g(\Omega) \otimes f'(\Omega)$
where $\otimes$ indicates convolution and $f'(\Omega)$ is the derivative of $f(\Omega)$.

$$= c u_{[\Omega_o,\, \Omega_{N+1}]}(\Omega) \otimes f'(\Omega)$$

$$+ \{g(\Omega) - c u_{[\Omega_o,\, \Omega_{N+1}]}(\Omega)\} \otimes f'(\Omega)$$

$$= c\,\{f(\Omega - \Omega_o) - f(\Omega - \Omega_{N+1})\} + \{\int_{\Omega_o}^{\Omega} [g(\Omega') - c]\, d\Omega'\} \otimes f''(\Omega).$$

Now

$$|\text{Arg.}\,\{Z(i\omega)\} - \theta| \leq c\,|\, f(\Omega - \Omega_o) - f(\Omega - \Omega_{N+1}) - \frac{\theta}{c}|$$
$$+ ||\int_{\Omega_o}^{\Omega} [g(\Omega') - c]\, d\Omega'\,||_\infty\, ||\, f''(\Omega)\,||_1$$

where $||\cdot||_\infty$ and $||\cdot||_1$ denote the norms in L_∞ and L_1 respectively.

Since $||\int_{\Omega_o}^{\Omega} [g(\Omega') - c]\, d\Omega'\,||_\infty = c(1-c)\log\rho$ and

$$||\, f''(\Omega)\,||_1 = \int_{-\infty}^{\infty} |f''(\Omega)|\, d\Omega \leq 2,$$

$$|\text{Arg.}\,\{Z(i\omega)\} - \theta| \leq c\;|\tan^{-1}\left(\frac{\omega}{\omega_o}\right) - \tan^{-1}\left(\frac{\omega}{\omega_{N+1}}\right) - \frac{\pi}{2}| + 2c(1-c)\log\rho.$$

Since we can always choose α, ρ and N so that for any ω in $[a,b]$,
$\tan^{-1}\left(\frac{\omega}{\omega_o}\right) - \frac{\pi}{2} < \frac{\varepsilon}{3}$, $\tan^{-1}\left(\frac{\omega}{\omega_{N+1}}\right) < \frac{\varepsilon}{3}$ and $2c(1-c)\log\rho < \frac{\varepsilon}{3}$, the inequality $|\text{Arg.}\,\{Z(i\omega)\} - \theta| < \varepsilon$ can be satisfied proving Lemma 3.1.

4.MAIN RESULTS

The following three theorems are direct consequencies of theorem 3.1 and Lemma 3.1, and constitute the main results of this paper.

Theorem 4.1: If the Nyquist plot of the linear part of the system $G(i\omega)$ lies entirely to the right of a straight line through $(-\frac{1}{K_2}, 0)$, the nonlinear feedback system in Fig.1 with $F(\cdot)$ satisfying $0 < \frac{F(y_1)-F(y_2)}{y_1 - y_2} < K_2$ in the feedback path is asymptotically stable (Fig.2).

Theorem 4.2: If the Nyquist plot $G(i\omega)$ lies entirely outside a circle passing through $(-\frac{1}{K_2},0)$ and $(-\frac{1}{K_1},0)$ (without completely encircling it), the nonlinear feedback system is asymptotically stable for all $F(\cdot)$ satisfying $K_1 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < K_2$ where $0 \leq K_1 < K_2 < \infty$.

Theorem 4.3: If the Nyquist plot $G(i\omega)$ lies entirely inside a circle passing through $(-\frac{1}{K_2},0)$ and $(\frac{1}{K_1}, 0)$, the system is asymptotically stable for all nonlinear functions $F(\cdot)$ satisfying $-K_1 < \frac{F(y_1)-F(y_2)}{y_1 - y_2} < K_2$ where K_1 and K_2 are positive constants.

Proof: Theorem 4.1

(i) Case A. $\pi/2 > \theta > 0$.

Since the locus of $G(i\omega)$ lies to the right of a straight line passing through $(-\frac{1}{K_2},0)$ the argument of $G_1(i\omega)$ satisfies the inequality $-\pi/2 - \theta < \text{Arg. } G_1(i\omega) < \pi/2 - \theta$, where $G_1(i\omega) = G(i\omega) + \frac{1}{K_2}$. Hence $|\text{Arg. } G_1(i\omega) + \theta| \leq \pi/2 - \delta$ where $\delta > 0$, $|\theta| < \pi/2$ and θ is the angle made by the straight line with the imaginary axis in Fig.2a. The phase vs. frequency plot of such a function $G_1(i\omega)$ for $\theta > 0$ may be represented as shown in Fig.3. But by lemma 3.1, there exists a multiplier $Z_{RL}(i\omega)$ (or a $Z_{RC}(i\omega)$ if $\theta < 0$) so that $|\text{Arg.}Z_{RL}(i\omega) - \theta| \leq \varepsilon$ for $a \leq \omega \leq b$ where $\varepsilon < \delta$. Hence one can always construct a $Z_{RL}(s)$ such that $|\text{Arg.}Z_{RL}G_1(i\omega)| \leq \pi/2$.

Hence $\text{Re } G_1(i\omega) Z_{RL}(i\omega) > 0$ and by theorem 3.1, the system is asymptotically stable.

(ii) Case B. $0 > \theta > -\pi/2$.

Since $\theta < 0$, one has to construct a $Z_{RC}(s)$ such that

$$|\text{Arg. } G_1 Z_{RC}(i\omega)| \leq \pi/2.$$

Again by lemma 3.1 there exists a $Z_{RC}(s)$ such that $\text{Re } G_1(i\omega)Z_{RC}(i\omega) \geq 0$. Hence by theorem 3.1, the system is again asymptotically stable.

Theorem 4.2: If the Nyquist plot $G(i\omega)$ lies entirely outside a circle passing through the points $(-\frac{1}{K_1},0)$, $(-\frac{1}{K_2},0)$ on the negative real axis (Fig.2b), the following transformations may be carried out:

a) Shifting the origin of coordinates to $(-\frac{1}{K_2},0)$ yields $G(i\omega)+\frac{1}{K_2}$ lying entirely outside a circle through the origin.

b) Inverting with respect to the origin, $\frac{K_2}{1+K_2G(i\omega)}$, lies entirely to the right of a straight line through $(\frac{-K_2K_1}{K_2-K_1},0)$.

Hence using the results of theorem 4.1, $\frac{K_2}{1+K_2G(i\omega)}$ is asymptotically stable for all nonlinear monotonic feedback functions in the sector

$$0 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < \frac{K_2 - K_1}{K_2K_1}$$

or $G(i\omega)$ is asymptotically stable for all monotonic functions $F(\cdot)$ satisfying $K_1 < \frac{F(y)}{y} < K_2$.

Theorem 4.3: The proof follows along the lines indicated for the previous cases.

5. SOME COMMENTS ON THE CIRCLE CRITERION

Some comments can now be made regarding the applicability of the criterion in practical situations as well as its relation to existing criteria.

(i) The circle criterion stated in theorem 4.2 and 4.3 may be applied in a practical situation in many ways. In particular the maximum range of stability given by the criterion may be determined as follows. If the lower limit of the monotonic gain K_1 is specified, the upper limit K_2 of the gain defined by $K_1 < \frac{F(y)}{y} < K_2$ within which the nonlinear feedback system is asymptotically stable may be obtained by drawing the circle, which passing through $(-\frac{1}{K_1},0)$ touches $G(i\omega)$ at two points. The second intersection of the circle with the negative real axis yields $-\frac{1}{K_2}$.

(ii) The criterion given by theorem 4.2 is a sufficient condition and by no means necessary for the stability of the nonlinear feedback system. Since the theorem is proved for the most stringent case, the stability range may be $K_1 < \frac{F(y)}{y} < \bar{K}_2$, $K_2 < \bar{K}_2$, in an actual situation for a specific $G(s)$. In such cases a multiplier Z_{RL} may exist so that $G(s)Z_{RL}(s)$ is positive real even when $G(i\omega)$ does not lie to the right of a straight line (or outside a circle).

(iii) If the lower limit $K_1 = 0$, for a given linear system $G(i\omega)$, gains K_N, K_M, K_P and K_C represent the gains for which the feedback system is asymptotically stable with (a) constant gain (K_N = Nyquist gain)

(b) monotonic nonlinearities (K_M obtained by the criterion stated in this paper) (c) first and third quadrant nonlinearities (K_P - Popov gain) and (d) nonlinear time-varying gains (K_C obtained by the circle criterion [5,6]) in the feedback path. The methods for obtaining these values in practical situations are illustrated in Figs.4,5 for examples 1 and 2. It is well known that $K_C \leq K_P \leq K_N$. In most situations the gain K_M is found to lie between the Nyquist gain and the Popov gain i.e. $K_P < K_M < K_N$.

(iv) The circle criterion for time varying systems calls for a circle with its center on the negative real axis while the circle criterion for monotonic nonlinearities calls for any circle touching $G(i\omega)$ at two points. The two are identical only in the particular case where $G(i\omega)$ is symmetrical about the real axis. In general the monotonic circle (C_M) criterion yields a much larger sector of stability than the circle (C_C) criterion for time varying systems. The sectors for a specific example are shown in Fig.5.

6.EXAMPLES

Two examples are given in this section to demonstrate the applicability of the criterion derived in section 5 and to compare it with Popov's criterion and the circle criterion for time-varying systems.

Example 1: The transfer function of the linear part of the feedback system in Fig. 1 is given by

$$G(s) = \frac{s+1}{s(s+0.1)(s^2+0.5s+9)} \, .$$

The Nyquist plot $G(i\omega)$ as well as the modified Nyquist plot (ReG vs. ωImG) are shown in Fig.4. The Nyquist gain for the system is $K_N = 4.17$ while the Popov gain $K_P = .63$ is obtained by drawing a straight line tangential at two points to the modified Nyquist plot. The straight line tangential to $G(i\omega)$ at two points intersects the real axis at $(-\frac{1}{3.23},0)$ yielding a $K_M = 3.23$. Hence the system is asymptotically stable for all monotonic increasing nonlinear functions in the sector $0 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < 3.23$.

Example 2: In this case

$$G(s) = \frac{3(s+1)}{s^2(s^2+s+25)}$$

It is required to find a sector $1 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < K_2$ within which the system is asymptotically stable. The Nyquist plot $G(i\omega)$ is shown in Fig.5.

(i) The Nyquist gain $K_N = 8.30$. Hence for all constant gains K in the feedback path $0 < K < 8.3$ the system is asymptotically stable.

(ii) A circle C_C with its center on the negative real axis, passing through (-1,0) and touching $G(i\omega)$ intersects the real axis a second time at $-\frac{1}{2.22}$. Hence for all nonlinear time-varying gain functions in the feedback path satisfying $1 < \frac{F(y,t)}{y} < 2.22$, the system is asymptotically stable.

(iii) A circle C_M passing through (- 1,0) and touching the Nyquist plot at two points intersects the real axis at $-\frac{1}{6.66}$. Hence by the circle criterion developed in this paper, the feedback system with monotonic increasing nonlinear functions in the feedback path is stable for all $F(\cdot)$ satisfying

$$1 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < 6.66$$

7. CONCLUSION

A simple geometrical criterion is derived in this paper as a sufficient condition for determining the asymptotic stability of a feedback system with a single monotonic increasing nonlinear gain function in the feedback path. It is shown that if the Nyquist plot $G(i\omega)$ of the linear part lies entirely outside a circle intersecting the negative real axis at $(-\frac{1}{K_1},0)$ and $(-\frac{1}{K_2},0)$, then the nonlinear system is asymptotically stable in the sector $K_1 < \frac{F(y_1) - F(y_2)}{y_1 - y_2} < K_2$.

Acknowledgement

The authors would like to thank the reviewer for suggesting the form of the multiplier used in the Main Lemma.

References:

1. V.M. Popov, "Absolute Stability of Nonlinear Systems of Automatic Control," Automation and Remote Control, pp. 857-875, March 1962.

2. (a) G. Zames, "On the Stability of Nonlinear, Time-varying Feedback Systems," Proc. NEC, Vol. 20, pp. 725-730, October 1964.

   (b) G. Zames, "On the Input-Output Stability of Time-varying Nonlinear Feedback Systems," pts. I and II, IEEE Trans. on Automatic Control, Vol. AC-11, pp.228-238, April 1966 and pp. 465-476, July 1966.

3. R.W. Brockett and J.W. Willems, "Frequency Domain Stability Criteria," pts. I and II, IEEE Trans. on Automatic Control, Vol. AC-10, pp. 255-261, July 1965 and pp. 407-413, October, 1965.

4. Kumpati S. Narendra and C.P. Neumann, "Stability of a Class of Differential Equations with a Single Monotone Nonlinearity," J.Siam Control, Vol. 4, pp. 295-308, May 1966.

5. Kumpati S. Narendra and R.M. Goldwyn, "A Geometrical Criterion for the Stability of Certain Nonlinear Nonautonomous Systems," IEEE Trans. on Circuit Theory, Vol. CT-11, pp. 406-408, September 1964.

6. (a) I.W. Sandberg, "On the L_2- Boundedness of Solutions of Nonlinear Functional Equations," Bell Sys.Tech. J., Vol. 43,pp.1581-1599, July

   (b) I.W. Sandberg, "A Frequency-Domain Condition for the Stability of Feedback Systems Containing a Single Time-varying Nonlinear Element," Bell Sys. Tech. J., Vol. 43, pp. 1601-1608, July 1964.

7. M.A. Aizerman and F.R. Gantmacher, "Absolute Stability of Regulator Systems," Holden-Day, Inc., San Francisco, Calif., 1964.

8. C.T. Lee and C.A. Desoer, "Stability of Single-Loop Nonlinear Feedback Systems," Rpt. ERL 66-13, Elec. Res. Lab., University of California, Berkeley, Calif., May 1966.

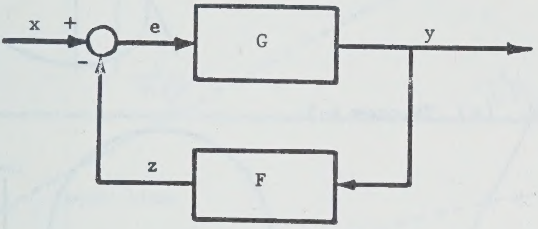


Fig.1 Nonlinear Feedback Systems

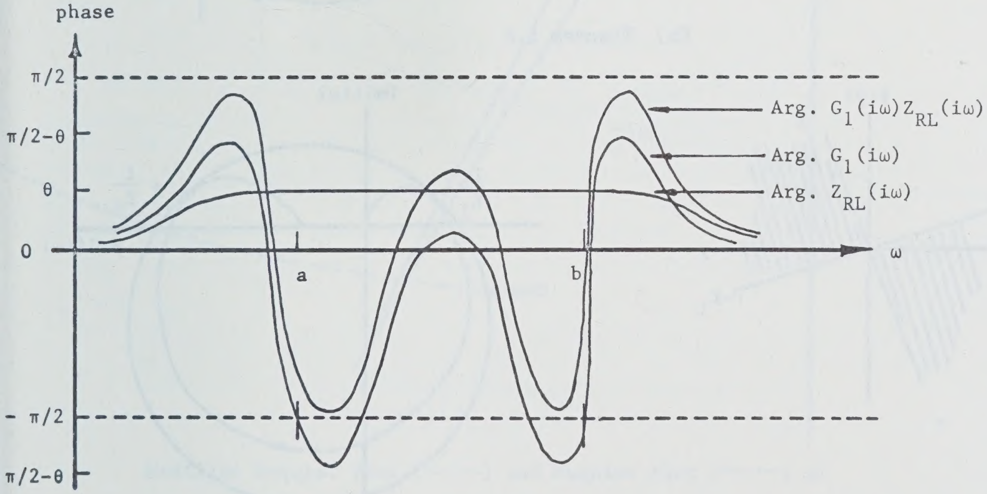
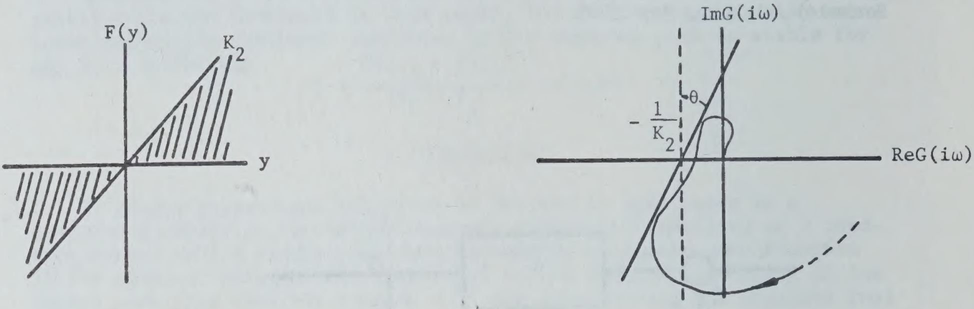
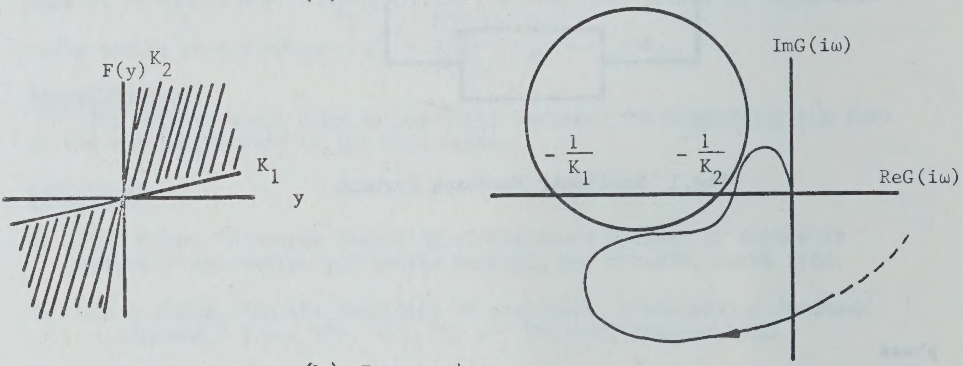


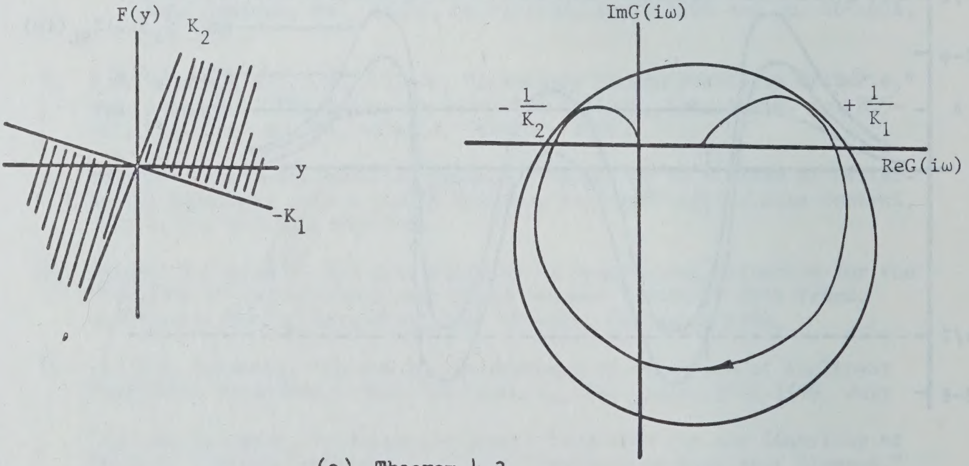
Fig.3 Proof of Theorem 4.1


F(y)
K_2
y
ImG(iω)
θ
$-\frac{1}{K_2}$
ReG(iω)

(a) Theorem 4.1

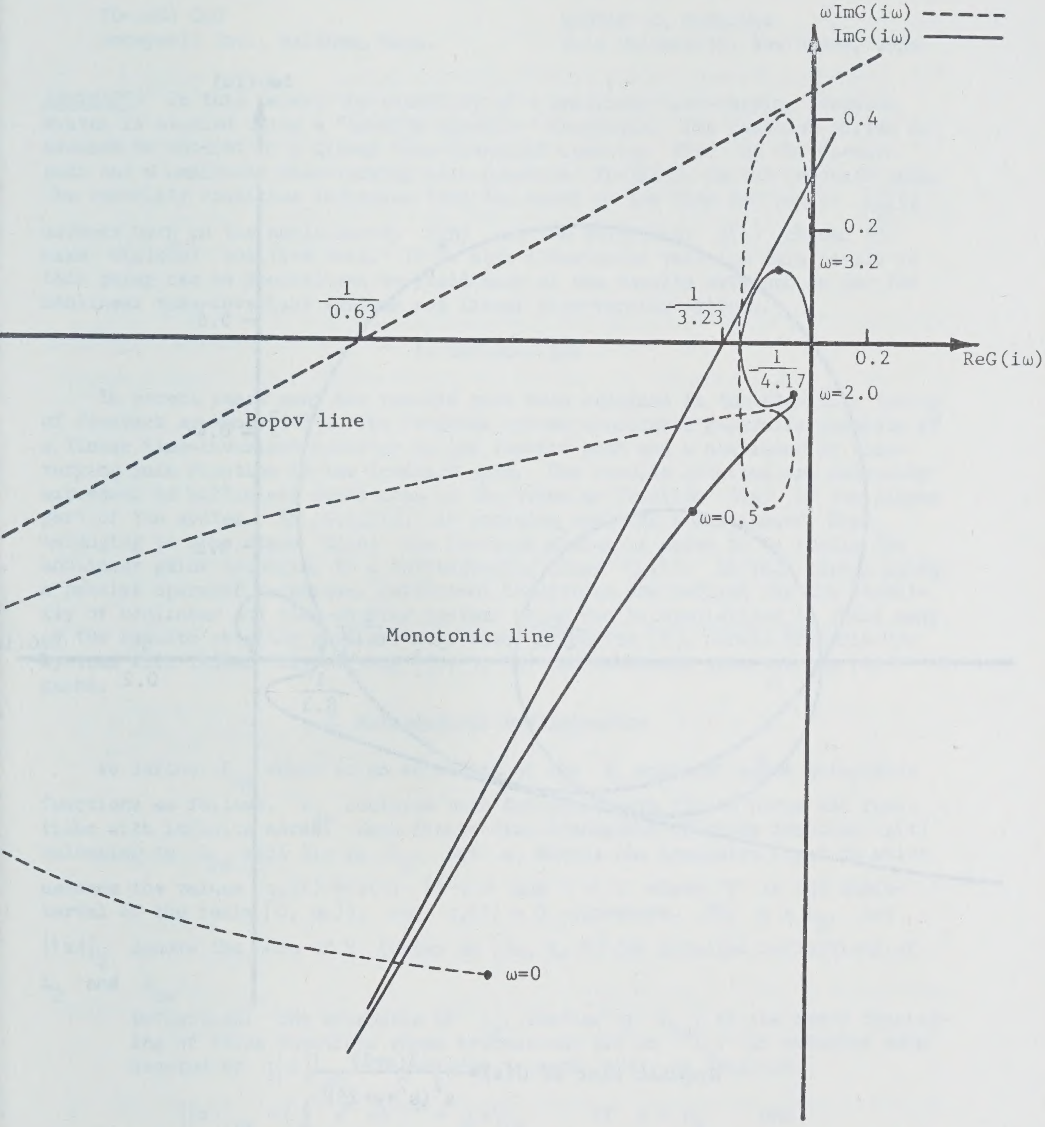

F(y)
K_2
K_1
y
ImG(iω)
$-\frac{1}{K_1}$
$-\frac{1}{K_2}$
ReG(iω)

(b) Theorem 4.2


F(y)
K_2
y
$-K_1$
ImG(iω)
$-\frac{1}{K_2}$
$+\frac{1}{K_1}$
ReG(iω)

(c) Theorem 4.3

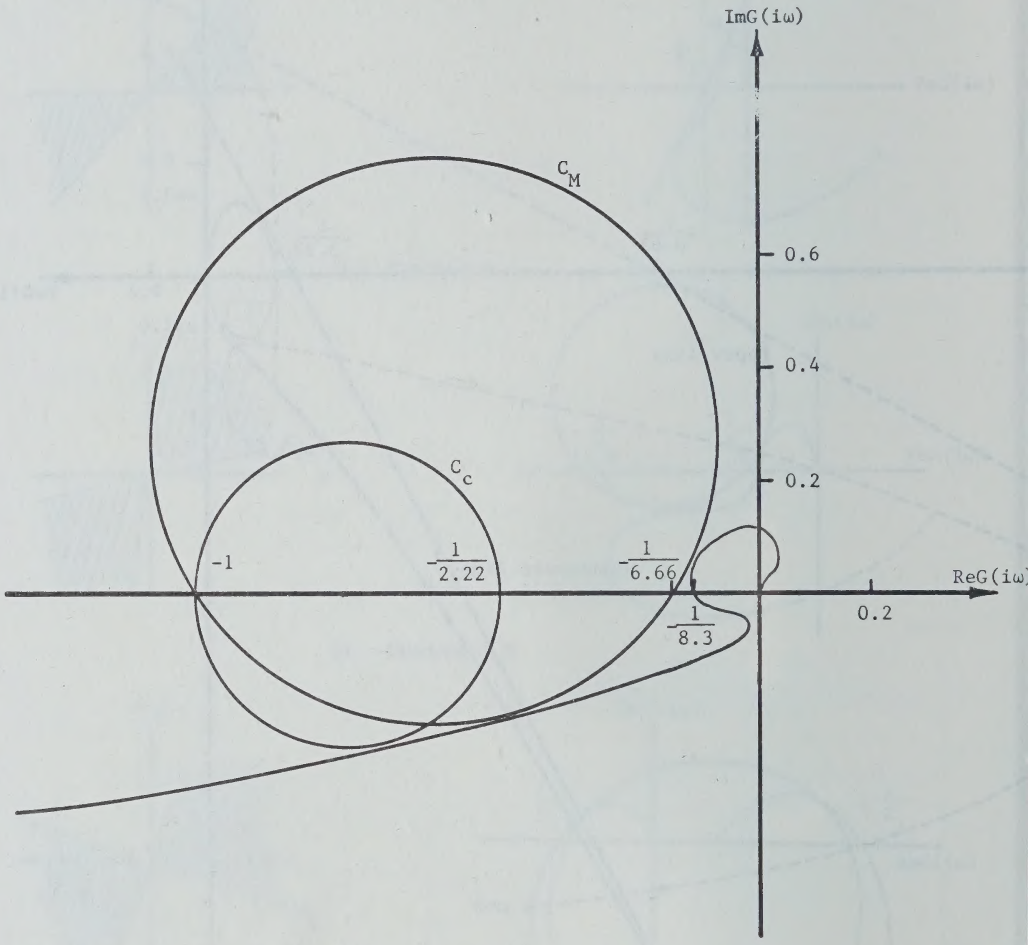
Fig.2 Admissible Regions for F(·) from the given G(iω)



Modified Nyquist Plot (-----) and Nyquist Plot (——) of

$$G(s) = \frac{(s+1)}{s(s+0.1)(s^2+0.5s+9)}$$

Fig.4 Example 1



Nyquist Plot of $G(s) = \frac{3(s+1)}{s^2(s^2+s+25)}$

Fig.5 Example 2

STABILITY OF NONLINEAR TIME-VARYING FEEDBACK SYSTEMS

YO-SUNG CHO
Honeywell Inc., Waltham, Mass.

KUMPATI S. NARENDRA
Yale University, New Haven, Conn.

ABSTRACT: In this paper, the stability of a nonlinear time-varying feedback system is studied using a "passive operator" technique. The feedback system is assumed to consist of a linear time-invariant operator $G(s)$ in the forward path and a nonlinear time-varying gain function $F(\cdot)K(t)$ in the feedback path. The stability condition indicates that the bound on the time derivative $\frac{dK(t)}{dt}$ depends both on the nonlinearity $F(\cdot)$ and the multiplier $Z(s)$ chosen to make $G(s)Z(s)$ positive real. It is also illustrated that the main result in this paper can be specialized to yield many of the results obtained so far for nonlinear time-invariant systems and linear time-varying systems.

1. INTRODUCTION

In recent years many new results have been obtained in the stability theory of feedback systems [1-8]. The feedback system considered generally consists of a linear time-invariant operator in the forward path and a nonlinear or time-varying gain function in the feedback path. The results obtained are generally expressed as sufficient conditions on the transfer function $G(s)$ of the linear part of the system. If $G(s)Z(s)$ is positive real for a multiplier $Z(s)$ belonging to some class $Z_c(s)$ the feedback system is shown to be stable for nonlinear gains belonging to a corresponding class $f_c(\cdot)$. In this paper, using a passive operator technique, sufficient conditions are derived for the stability of nonlinear and time-varying systems which can be specialized to yield many of the results obtained earlier, e.g. Popov criterion [1], circle criteria for systems with linear time-varying [3c, 5, 8a] and nonlinear time-varying [4a] gains.

2. Mathematical Preliminaries

We define L_{2e} which is an entension of the L_2 space of square integrable functions as follows. L_{2e} contains both functions with finite norms and functions with infinite norms. Each finite time truncation of every function $x(t)$ belonging to L_{2e} will lie in L_2. Let x_τ denote the truncated function which assumes the values $x_\tau(t) = x(t)$ $(t < \tau$ and $\tau \in T$ where T is the subinterval of the reals $[0, \infty))$, and $x_\tau(t) = 0$ elsewhere. For $x \in L_2$, let $||x||_2$ denote the norm of x (Refer to [3a, 4, 6] for detailed definitions of L_2 and L_{2e}).

Definition: The extension of L_2, denoted by L_{2e}, is the space consisting of those functions whose truncations lie in L_2. An extended norm denoted by $||x||_{2e}$ is assigned to each $x(t)$ as follows.

$$||x||_{2e} = \left\{ \int_0^\infty x^2 \, dt \right\}^{\frac{1}{2}} = ||x||_2 \qquad \text{if } x \in L_2 \quad \text{and}$$

$$||x||_{2e} = \infty \qquad \text{if } x \notin L_2 \;.$$

Assuming G and F in Fig. 1a map L_{2e} into itself, one may formulate feedback equations in L_{2e} for the given system as follows;

Let $G(e)$ indicate the operation of G on an input function $e(t)$, then

$$\left.\begin{aligned} G(e) &= y(t) \\ F(G(e)) + e(t) &= x(t) \end{aligned}\right\} \qquad (1)$$

The objective of this paper is to find sufficient conditions on G and F as defined below under which $x(t) \in L_2$ and $e(t), y(t) \in L_{2e}$ implies $e(t), y(t) \in L_2$ and $y(\infty) = 0$.

2.1 Classes of Nonlinear Time-Varying Gains $F(\cdot, t)$

Let $F(\cdot, t)$ be a no-memory nonlinear time-varying real valued function of the real variables and satisfy the following conditions.

(i) $F(\cdot, t)$ is expressed as the product of a no-memory nonlinear function $f(\cdot)$ and a linear time-varying gain $K(t)$ such that $F(\cdot, t) = K(t)\, f(\cdot)$.

(ii) $F(\sigma, t) = 0$ if and only if $\sigma(t) = 0$.

(iii) Let F satisfy the conditions (i) and (ii), and let $0 < \frac{F(\sigma, t)}{\sigma} \leq K_2 < \infty$ for all $0 \leq t < \infty$ and $\sigma(t) \neq 0$ where K_2 is an arbitrary large positive real constant. Then $F \in F_p$.

(iv) Let F satisfy the conditions (i) and (ii), and let $0 < \frac{F(\sigma, t)}{\sigma} < K_2$ for all $0 \leq t < \infty$ where K_2 is a finite positive real constant expressed by $K_2 = K_t \cdot K_N$ where $0 < K(t) < K_t$ and $0 < \frac{f(\sigma)}{\sigma} < K_N$. Then $F \in F_{pk}$.

(v) Let F satisfy the conditions (i), (ii) and (iii), and let

$$\left\{ F(\sigma_1, t) - F(\sigma_2, t) \right\} \cdot \left\{ \sigma_1 - \sigma_2 \right\} \geq 0 \qquad (2)$$

for all $0 \leq t < \infty$. Then $F \in F_m$. If $F \in F_m$, it is assumed that $F^{-1}(\cdot, t)$ exists and belongs to the class F_m.

(vi) Let F satisfy the conditions (v), and $0 < \frac{F(\sigma, t)}{\sigma} < K_2$ for all $0 \leq t < \infty$ where K_2 is a finite positive real constant expressed by $K_2 = K_t \cdot K_N$ where $0 < K(t) < K_t$ and $0 < \frac{f(\sigma)}{\sigma} < K_N$, then $F \in F_{mk}$. If $F \in F_{mk}$, it is noted that $0 < \frac{F(\sigma_1, t) - F(\sigma_2, t)}{\sigma_1 - \sigma_2} < \infty$.

(vii) Let F satisfy the conditions (v) and the condition $F(\sigma, t) = -F(-\sigma, t)$ for all $0 \leq t < \infty$, then $F \in F_{om}$. In a similar manner, the class F_{omk} may also be defined.

2.2 Linear Time-Invariant Operator G

From a given feedback system (Fig. 1a), the operator G is linear time-invariant and is assumed to satisfy the following condition.

$$G(e) = \int_0^t g(t-\xi)\, e(\xi)\, d\xi + g_i(t) \qquad (3)$$

where $g(t) \in L_1 \cap L_2$ is a unit impulse response of G and $g_i(t) \in L_2$ is the result of initial conditions.

By a simple transformation, a new feedback equation without initial conditions may be obtained as follows.

$$\int_0^t g(t-\xi)\, F\left[y(\xi)\right] d\xi + y(t) = x^*(t) \qquad (4)$$

where the new input function satisfies the equation $x^*(t) = \int_0^t g(t-\xi)\, x(\xi)\, d\xi + g_i(t)$. Since $x(t), g_i(t) \in L_2$ implies $x^*(t) \in L_2$, the initial condition in the loop can be assumed to be zero without any loss of generality.

2.3 Passive Operators

If $x, y \in L_2$, an inner product in L_2 is defined by setting $\langle x, y \rangle = \int_0^\infty x(t)\, y(t) dt$. Since every truncated function $x_\tau(t), y_\tau(t) \in L_2$, if $x(t), y(t) \in L_{2e}$, we can define the following notation;

$$\langle x, y \rangle_\tau = \langle x_\tau(t), y_\tau(t) \rangle = \int_0^\infty x_\tau(t)\, y_\tau(t)\, dt$$
$$= \int_0^\tau x(t)\, y(t)\, dt \qquad \text{where } \tau \in T .$$

i) Passive operator: Let an operator S map L_{2e} into itself and $x(t) \in Do(S)$. If $x(t) \in L_{2e}$ and $\langle x, S(x) \rangle_\tau \geq 0$ for all $\tau \in T$, the operator S is said to be passive.

ii) δ-passive operator: If there exists a positive real number δ such that $\langle x, S(x) \rangle_\tau \geq \delta \langle x, x \rangle_\tau$, the operator S is δ-passive.

iii) δ-M passive operator: If $\langle x, S(x) \rangle_\tau \geq \delta \langle S(x), S(x) \rangle_\tau$, S is a δ-M passive operator.

2.4 Basic Theorems

The results derived in Sections 3 and 4 are direct consequences of the following two basic stability theorems and Lemma 2.1.

Theorem 2.1: In the feedback system in Fig. 1a, if the operators G and F map L_{2e} into itself and F is δ-M passive or δ-passive such that $\langle F(y), y \rangle_\tau \geq \delta \langle F(y), F(y) \rangle_\tau$ or $\langle F(y), y \rangle_\tau \geq \delta \langle y, y \rangle_\tau$ respectively ($\tau \in T$), then the system is L_2-stable (i.e. $y \in L_2$ when $x \in L_2$) provided that G is a $-\epsilon$ passive or $-\epsilon$ M passive operator respectively (i.e. $\langle G(e), e \rangle_\tau \geq -\epsilon \langle e, e \rangle_\tau$ or $\langle G(e), e \rangle_\tau \geq -\epsilon \langle G(e), G(e) \rangle_\tau$ where δ and ϵ are positive real numbers and $\delta > \epsilon \geq 0$) . The converse statement resulting from interchanging operators F and G is also true. (Similar results are found in [3a, 4b, 6]) .

Proof: Case 1: F is δ-M passive and G is $-\epsilon$ passive. Referring to Fig.1, the system equation in L_{2e} is $x = [I + FG]\, e$.

$$x = [I + FG]\, e \quad . \tag{5}$$

The objective is to find the space of e when $x \in L_2$. Multiplying both sides of the system equation by 2δ ,

$$\begin{aligned} 2\delta x &= [2\delta I + 2\delta FG]\, e \\ &= [2\delta I + G] e + [2\delta F - I]\, G\, [2\delta I + G]^{-1} [2\delta I + G] e \quad . \end{aligned} \tag{6}$$

Defining $x_1 = 2\delta x \in L_2$ and $e_1 = [2\delta I + G]\, e \in L_{2e}$ if $e \in L_{2e}$, Eq. (6) becomes

$$x_1 = e_1 + [2\delta F - I]\, G\, [2\delta I + G]^{-1}\, e_1 \tag{7}$$

Defining $F_1 = 2\delta F - I$ and $G_1 = G\, [2\delta I + G]^{-1}$, it can be shown that $\langle G_1(e_1), G_1(e_1) \rangle_\tau \geq a \langle e_1, e_1 \rangle_\tau$ (where $a < 1$ is a positive real constant), and $\langle F_1(e_2), F_1(e_2) \rangle_\tau \leq \langle e_2, e_2 \rangle_\tau$ if the hypotheses of Theorem 2.1 are satisfied. Since $F_1 G_1$ is a contraction operator in L_{2e} space (i.e. L_{2e} norm of $F_1 G_1$ is less than one), $e_1 \in L_2$ if $x \in L_2$ from Eq. (7) (See the definition of contraction operator in Ref. 4). Hence $y \in L_2$ and $e \in L_2$.

Case 2: G is δ-M passive and F is $-\epsilon$ passive. Operating on both sides of the system equation with G, the equation $x_1 = e_1 + GF\, e_1$ is obtained where $x_1 = Gx$ and $e_1 = G\, e$. This reduces to the equiv-system equation (5) of Case 1.

Case 3: F is δ-passive and G is $-\epsilon$ M passive.

$$\begin{aligned} x &= e + F\, Ge \\ &= [I + \delta G]\, e + [F - \delta I]\, G [I + \delta G]^{-1} [I + \delta G]\, e \end{aligned}$$

$$= e_1 + F_1 G_1 e_1 \qquad (8)$$

where $e_1 = [I + \delta G] e$, $F_1 = [F - \delta I]$, $G_1 = G[I + \delta G]^{-1}$.

Then if the conditions of Theorem 2.1 are satisfied,

$$< G_1(e), e >_\tau \geq (\delta - \epsilon) < G_1(e), G_1(e) >_\tau \quad \text{and} \quad < F_1 e_2, e_2 >_\tau \geq 0 .$$

Hence, the Eq.(8) reduces to the Case 1.

Case 4: G is δ- passive and F is $-\epsilon$ M passive.

The proof follows along the lines indicated in Cases 2 and 3.

Theorem 2.2: If the feedback system is L_2 stable and F and G belong to the classes stated in Sections 2.1 and 2.2, then $y(t) \epsilon L_2 \cap L_\infty$ and $y(\infty) = 0$ (i.e. system is asymptotically stable) .

Proof: The proof is omitted here. A similar result is also found in [3a, 8b] .

2.4 Lemmas

Let linear time-invariant operators Z_N, Z_{RL}, Z^*_{RLC} be Laplace transformable and have the forms:

i) $Z_N(s) = s + \alpha$ where $\alpha \geq 0$

ii) $Z_{RL}(s) = \prod_{i=1}^{N} \frac{s + \alpha_i}{s + \beta_i}$ where $0 \leq \alpha_1 < \beta_1 < \alpha_2 < \beta_2 < \cdots$. and

iii) $Z^*_{RLC}(s) = \alpha_o + \beta_o s + \sum_{i=1}^{\ell} \epsilon_i \frac{s + c_i \gamma_i}{s + \gamma_i} + \sum_{j=1}^{m} \delta_j \frac{s + \zeta_j}{s^2 + \eta_j s + \lambda_j} + \sum_{k=1}^{n} \phi_k \frac{s^2 + \mu_k s + \nu_k}{s^2 + \xi_k s + \rho_k}$

where all the coefficients are non-negative,

$$0 \leq c_i \leq 2 , \quad \sum_{j=1}^{m} \alpha_j = \alpha_o > 0 , \quad \eta_j - \zeta_j - \frac{\lambda_i}{2\zeta_j} \geq \frac{\delta_j}{2\alpha_j} , \quad \xi_k \geq \mu_k , \quad \rho_k \geq \nu_k$$

and $\mu_k - \frac{\rho_k - \nu_k}{\xi_k - \mu_k} - \frac{\nu_k}{2} \frac{(\xi_k - \mu_k)}{(\rho_k - \nu_k)} \geq 0$, (the form of $Z^*_{RLC}(s)$ appeared in [8b] and is used to prove the stability of systems containing an odd monotonic function),

and let Z_p, Z_m, Z_{om} be defined as

i) $Z_p^{-1}(s) = Z_N(s+\beta)$

ii) $Z_m^{-1}(s) = Z_{RL}(s+\beta)$

iii) $Z_{om}^{-1}(s) = Z^*_{RLC}(s+\beta)$ where β is a positive real constant.

Then, the following Lemma 2.1 may be stated.

Lemma 2.1: i) If a nonlinear time-varying function $F(\cdot, t) \epsilon F_p$ is preceded by a linear operator Z_p , then $F[Z_p(\cdot), t]$ is a δ-M passive operator provided that $\frac{\dot{K}}{K} < \beta F_{min}$ where $\dot{K} = \frac{dK(t)}{dt}$, $F_{min} = \min_{w(t)} \frac{f(w)\, w}{\int_0^w f(v)\, dv}$ and $0 < F_{min} < \infty$.

ii) Similarly, if $F(\cdot, t) \epsilon F_m$, $F[Z_m(\cdot), t]$ is a δ-M passive operator provided that $\frac{\dot{K}}{K} < \beta F_{min}$,($1 \leq F_{min} < \infty$) .

iii) If $F(\cdot, t) \epsilon F_{om}$, $F[Z_{om}(\cdot), t]$ is δ-M passive provided that $\frac{\dot{K}}{K} < \beta F_{min}$,($1 \leq F_{min} < \infty$) .

Proof: Case i) Let x and y be in L_{2e} and defined by the relation $y(t) = K(t)\, f[Z_p(x)]$. Then, it must be shown that $< x, y >_\tau \geq \delta < y, y >_\tau$ where $\delta > 0$ and $\tau \epsilon T$.

If $w(t)$ is defined by the equation $\dot{w} + \beta w = x$, $Z_p(x) = w$ and $y = K(t)\, f(w)$. Hence,

$$< x, y >_\tau \; = \; < \dot{w} + \beta w, Kf(w) >_\tau \; . \qquad (9)$$

If $\beta = \beta_1 + \epsilon$ where ϵ is an arbitrary small positive number

$$< x, y >_\tau \; = K(t) \left\{ \int_0^w f(v)dv \right\} \Big|_0^\tau - \int_0^\tau \dot{K} \left\{ \int_0^w f(v)dv \right\} dt + \beta_1 \int_0^\tau Kf(w)wdt + \epsilon \int_0^\tau Kf(w)wdt \; . \qquad (10)$$

Since it can be assumed that there are no initial conditions in the loop as stated in Section 2.2, the first term of Equation (10) is greater than zero. If $\frac{\dot{K}}{K} \leq \frac{\beta_1 f(w)\, w}{\int_0^w f(v)dv}$ the second and third terms are together greater than or equal to zero. The fourth term is greater than zero and is in fact greater than or equal to $\frac{\epsilon}{K_2} \int_0^\tau \{ Kf(w) \}^2 \, dt$ if $0 < \frac{Kf(w)}{w} \leq K_2 < \infty$. Hence, $< x, y >_\tau \; \geq \frac{\epsilon}{K_2} < y, y >_\tau$.

Case ii) If $y(t) = K(t)\, f[Z_m(\cdot)]$, it must be shown that

$$< x, y >_\tau \; \geq \delta < y, y >_\tau \qquad (11)$$

where $\delta > 0$ and $\tau \epsilon T$.

Let $Z_m(s) = \prod_{i=1}^{N} \frac{(s + \gamma_i)}{(s + \eta_i)}$ where $0 < \eta_1 < \gamma_1 < \eta_2 < \gamma_2 < \ldots.$ and $\eta_1 \geq \beta$. Expanding $Z_m^{-1}(s)$ in partial fractions with respect to $s + \beta$,

$$Z_m^{-1}(s) = a_o + \beta_o (s+\beta) + \sum_{i=1}^{N} \delta_i \frac{s+\beta}{s+\gamma_i} \qquad (12)$$

$$= Z_{(1)}(s) + Z_{(2)}(s) + \sum_{i=1}^{N} Z_i(s)$$

where all coefficients are non-negative real numbers. Defining $w(t)$ by the relationship $x(t) = Z_m^{-1}(w)$, the l.h.s. of inequality (11) becomes

$$< Z_{(1)}(w), K(t)f(w) >_\tau + < Z_{(2)}(w), K(t)f(w) >_\tau + \sum_{i=1}^{N} < Z_i(w), K(t)f(w) >_\tau \; . \qquad (13)$$

The first term of (13) is obviously δ-M passive. By Case i), the second term of (13) is also δ-M passive if $\frac{\dot{K}}{K} < \beta F_{min}$. The third term can be shown to be positive as follows: Defining $u(t)$ by $\dot{u} + \gamma_i u = w(t)$, $Z_i(w) = \delta_i (\dot{u} + \beta u)$. Then,

$$< Z_i(w), K(t)f(w) >_\tau \; = \delta_1 < (\dot{u} + \beta u), \; K(t)f(\dot{u} + \gamma_i u) >_\tau \; . \qquad (14)$$

Subtracting $\delta_i < (\dot{u} + \beta_o u), K(t)f((\gamma_i - \beta_o)u) >_\tau$ (15)

from (14), one obtains a non-negative quantity by the monotonic property (2) of $F(\cdot)$. But the expression (15) is greater than zero if $\frac{\dot{K}}{K} < \beta F_{min}$ by case i). Hence, the expression (14) is greater than zero. Combining all these, one can show that $F[Z_m(\cdot), t]$ is δ-M passive.

Case iii): By employing the same methods as were used in the previous cases and Lemmas 2.3-2.5 of Ref. [8b], passivity of $F[Z_{om}(\cdot), t]$ can be proved. However, the proof is omitted here.

3. EXTENSION OF POPOV'S CRITERION TO NONLINEAR TIME-VARYING SYSTEMS

It is well known that if $\left[G(s) + \frac{1}{K_c}\right]$ is positive real, the feedback system in Fig. 1a is asymptotically stable for all

$$0 < \frac{F(\sigma, t)}{\sigma} < K_c \tag{16}$$

by the circle criterion [2, 3b]. Further, by Popov's criterion if a gain K_p is found so that $G(s)\left(\frac{s}{\alpha} + 1\right) + \frac{1}{K_p}$ is positive real, then $K_p \geq K_c$ and the nonlinear autonomous system is stable for all

$$0 < \frac{f(\sigma)}{\sigma} < K_p\ . \tag{17}$$

In this section stability criteria are derived for feedback systems with nonlinear time-varying gains. The results in this section can be specialized to yield the criteria of (16) or (17). In the general case, the criterion derived involves a trade-off between the rate of variation of $K(t)$ and the range of the nonlinear gain function $f(\cdot)$.

Main Theorem 3.1: If $F(\cdot, t) \in F_p$, G belongs to the class stated in 2.2 and $G(s)\, Z_p^{-1}(s)$ is a passive operator where $Z_p^{-1}(s) = \frac{s}{\alpha} + 1$ (α is a non-negative real number), then the system (Fig. 1a) is asymptotically stable if

$$\frac{\dot{K}}{K} < \alpha F_{min}(w)\ .$$

Proof: Since $G(s)\, Z_p^{-1}(s)$ is a passive operator by hypothesis, it is sufficient to show that $F[Z_p(\cdot), t]$ is a δ-M passive operator by Theorems 2.1 and 2.2. But the δ-M passivity of $F[Z_p(\cdot), t]$ is proved by Case i) of Lemma 2.1 if $\frac{\dot{K}}{K} < \alpha F_{min}(w)$.

Theorem 3.2: If $F(\cdot, t) \in F_{pk}$, G belongs to the class stated in 2.2 and $G(s)\, Z_p^{-1}(s) + \frac{1}{K_2}$ is a passive operator (passivity and positive realness are interchangeable in the linear time-invariant operator), then the system is asymptotically stable if

$$\frac{\dot{K}}{K} < \alpha F_{min} \cdot \left(1 - \frac{K(t)}{K_t}\right)\ . \tag{18}$$

Proof: It is sufficient to show that $F[Z_p(\cdot), t]$ is δ-M passive, or by Theorems 2.1 and 2.2,

$$< F[Z_p(x), t],\ x >_\tau\ \geq \frac{1}{K_2} < F[Z_p(x), t],\ F[Z_p(x), t] >_\tau\ , \quad (\tau \in T). \tag{19}$$

Defining $w(t)$ by the relation $\frac{\dot{w}}{\alpha} + w = x(t)$, $w(t) = Z_p(x)$ and $F[Z_p(x), t] = f(w)\, K(t)$. Hence, $< F[Z_p(x), t],\ x >_\tau\ = < \frac{\dot{w}}{\alpha} + w,\ f(w)\, K(t) >_\tau$

$$= \frac{1}{\alpha} K(t) \left\{ \int_0^w f(v)dv \right\}\Big|_0^\tau - \frac{1}{\alpha} \int_0^\tau \dot{K} \left\{ \int_0^w f(v)dv \right\} dt + \int_0^\tau K\left(1 - \frac{K}{K_t}\right) f(w)\, w\, dt$$

$$+ \int_0^\tau \frac{K^2(t)}{K_t} f(w)\, w\, dt\ .$$

The first term is greater than or equal to zero, the second and the third terms together are greater than or equal to zero if condition (18) is satisfied. The last term yields

$$\int_0^\tau \frac{K^2(t)}{K_t} f(w) w dt \geq \int_0^\tau \frac{K^2(t)}{K_t} \frac{f^2(w)}{K_N} dt = \frac{1}{K_2} \int_0^\tau \{ f(w)\, K(t) \}^2 dt$$

Hence inequality (19) is satisfied.

Comments on Theorems 3.1 and 3.2: The condition $\frac{\dot{K}(t)}{K(t)} < \frac{\alpha f(w)\ w}{\int_0^w f(v)dv} \cdot (1 - \frac{K(t)}{K_t})$ indicates that the rate of variation $\dot{K}(t)$ as indicated by $\frac{\dot{K}}{K}$ depends on the nonlinear function $f(\cdot)$ that is considered. For example, if $f(\cdot)$ is a purely linear constant gain (i.e. $F(\cdot, t) = K(t)$), the stability condition of Theorem 3.2 becomes $\frac{\dot{K}}{K} < 2\,\alpha(1 - \frac{K}{K_t})$, a result, which is known for linear time-varying systems [3c, 5, 8a] . If the entire class of nonlinear functions in the sector $0 < \frac{f(\sigma)}{\sigma} < K_2$ is considered, then $F_{min} \rightarrow 0$ (i.e. $\dot{K} \rightarrow 0$) or $F(\cdot, t)$ has to be a linear time-invariant nonlinear function, as in Popov's criterion [1] . However, if the specific nonlinear function exists such that $\frac{f(w)\ w}{\int_0^w f(v)\ dv}$ has a minimum value β (e.g. if $f(w) = w^3$, $\beta = 4$), then $\frac{\dot{K}}{K} < \alpha\beta(1 - \frac{K}{K_t})$. For the particular case, when $K_2 \rightarrow K_c$, $\alpha \rightarrow \infty$ (i.e. $G(s) + \frac{1}{K_2}$ becomes positive real) so that the system is asymptotically stable for all $0 < \frac{F(\sigma,\ t)}{\sigma} < K_c$ which is the range predicted by the circle criterion [2, 3b] .

4. STABILITY CRITERION FOR THE CLASSES F_m AND F_{om}

If the class of monotonic increasing nonlinear functions is considered, it is well known that the range of stability may be increased in many situations beyond the Popov sector to $0 < \frac{f(\sigma)}{\sigma} < K_M$ where $K_M \geq K_p$. The sufficient condition for stability in the sector is known to be $[G(s) + \frac{1}{K_M}]\cdot Z(s)$ is P.R. (positive real) [4b], where $Z(s)$ has the form of $Z_{RL}(s)$ or $Z_{RL}^{-1}(s)$. The extension of this criterion to nonlinear time-varying systems is considered in this section.

Theorem 4.1: i) If $F(\cdot, t) \epsilon F_M$, G belongs to the class stated in 2.2 and $G(s)Z_m^{-1}(s)$ is P.R. where $Z_m^{-1}(s)$ is defined in 2.4, then the system is asymptotically stable if $\frac{\dot{K}}{K} < \beta F_{min}$, where $F_{min} = \min_w \frac{f(w)\ w}{\int_0^w f(v)\ dv}$ and $1 \leq F_{min}(w) < \infty$ and $-\beta$ is the location of zero of $Z_m^{-1}(s)$ which is nearest to the origin.

ii) If $F(\cdot, t) \epsilon F_{om}$ and $G(s)Z_{om}^{-1}(s)$ is P.R., then the system is asymptotically stable if $\frac{\dot{K}}{K} < \beta F_{min}$.

Proof: i) Since $G(s)\cdot Z_m^{-1}(s)$ and $F[Z_m(\cdot), t]$ are passive and δ-M passive by the hypothesis and case ii) of Lemma 2.2 respectively, the system is asymptotically stable by Theorems 2.1 and 2.2.

ii) Same reasons as i).

It should be noted that the multiplier $Z_m(s)$ (or $Z_{om}(s)$) is found such that $G(s)\cdot Z_m(s)$ is P.R., it can be shown that one of the stability conditions is reversed so that $-\frac{\dot{K}}{K} < \beta F_{min}$.

Theorem 4.2: i) If $F(\cdot, t) \epsilon F_{mk}$ and G belongs to the class stated in 2.2, then the system is asymptotically stable if

$$\left[G(s)\ Z_m^{-1}(s) + \frac{\alpha}{K_2}\right] \tag{20}$$

is P.R. and

$$\frac{\dot{K}}{K} < \beta F_{min}(1 - \frac{K(t)}{K_t})\ ,\quad (1 < F_{min} < 2) \tag{21}$$

where $\alpha = \alpha_o + \beta_o \beta$ (see Eq. (12)) .

ii) If $F(\cdot, t) \in F_{omk}$ and all other conditions are the same as in i), the asymptotic stability condition corresponding to (20) is that $G(s) \cdot Z_{om}^{-1}(s) + \frac{\alpha}{K_2}$ be P.R. where $\alpha = \alpha_o + \beta_o \beta$ (α_o, β_o are the first and second coefficients of $Z_{om}^{-1}(s)$ defined in 2.4.) A result similar to case i) but more restrictive is derived in [4a] for the case of slope bounded nonlinearities and corresponds to the case when $F_{min} = 1$.

Proof: Case i) By Theorems 2.1 and 2.2, it is sufficient to show that

$$< F[Z_m(x), t], x >_\tau \geq \frac{\alpha}{K_2} < F[Z_m(x), t], F[Z_m(x), t] >_\tau \qquad (22)$$

Defining $u(t) = Z(x) \in L_{2e}$, one can obtain an inequality similar to (13). The l.h.s. of inequality (22) may be expressed in the form,

$$\alpha_o < u, K(t)f(u) >_\tau + < Z_{(2)}(u), K(t)f(u) >_\tau + \sum_{i=1}^{n} < Z_i(u), K(t)f(u) >_\tau . \qquad (23)$$

The first term is definitely greater than or equal to $\frac{\alpha_o}{K_2} < K(t)f(u), K(t)f(u) >_\tau$. The second term can be shown to be greater than or equal to $\frac{\beta\beta_o}{K_2} < K(t)f(u), K(t)f(u) >_\tau$ if inequality (21) is satisfied using the same procedure as in Theorem 3.2. The last term of (23) is greater than or equal to zero if $\frac{\dot{K}}{K} < \beta F_{min}(w)$ by Eq. (14). Hence, the inequality (22) is satisfied.

Case ii) Omitted here.

In Theorem 4.2, the value of β is directly dependent on $\frac{\dot{K}}{K}$. As $\dot{K} \to 0$, $\beta \to 0$. Hence, when the monotonic gain function lies in the range $0 < \frac{f(\sigma)}{\sigma} < K_2$ and is not slope bounded, the condition for asymptotic stability is that $G(s)Z(s) + \frac{\alpha_o}{K_2}$ be positive real, where $Z(s)$ has the form $Z_m^{-1}(s)$ or $Z_{om}^{-1}(s)$.

5. INTERPRETATION OF RESULTS

The results of Theorems 3.1, 4.1, and 4.2 yield sufficient conditions for the stability of feedback systems with nonlinear time-varying feedback gains having the general form $f(\cdot)K(t)$. These conditions have the same form as well established frequency domain conditions for nonlinear autonomous systems and are expressed in terms of the existence of a multiplier $Z(s)$ to make $G(s)Z(s)$ (or $G(s)Z(s) + \frac{1}{K_2}$ in the finite sector case) positive real. The results indicate that the same multiplier used in the nonlinear time-invariant case can also be used for the nonlinear time-varying case to prove stability provided $\frac{\dot{K}}{K}$ is bounded in some fashion.

The bound on $\frac{\dot{K}}{K}$ is seen to depend on two factors: γ and F_{min} where γ is the distance of the nearest singularity of $Z(s)$ from the imaginary axis and

$$F_{min} = \left. \frac{f(w)\, w}{\int_0^w f(v)dv} \right|_{min} .$$

Hence γ depends on the particular multiplier chosen and $F_{min}(w)$ on the particular nonlinear function included in the feedback path. The latter can be zero when $f(\cdot)$ includes the entire class of nonlinearities in the sector $0 < \frac{f(\sigma)}{\sigma} < K_N$ but is finite and lies in the interval $0 < F_{min} < \infty$ for a specific nonlinear function. For monotonic nonlinear functions F_{min} satisfies $1 \leq F_{min} < \infty$, indicating that the stability criterion derived for nonlinear time-invariant systems is also valid for time-varying cases for some slowly varying function $K(t)$.

Assuming $0 < K(t) < 1$ and $0 < \frac{f(\sigma)}{\sigma} < K_N = K_2$, it is well known that if $G(s)Z_1(s) + \frac{1}{K_2}$ is P.R. for some multiplier $Z_1(s)$, then a multiplier $Z_2(s)$ can be found so that $G(s)Z_2(s) + \frac{1}{K_2^*}$ is P.R., where $K_2^* \leq K_2$. If γ_1 and γ_2 are the distances of the nearest singularities of $Z_1(s)$ and $Z_2(s)$ from the imaginary axis, γ_2 can be made larger than γ_1. In such a case, the system is stable in the range $0 < f(\sigma)\, K(t) < K_2^* \sigma$ if $\frac{\dot{K}(t)}{K(t)} < \gamma_2\, F_{min}(w) \cdot (1 - K(t))$ so that the range of variations can be decreased to increase the rate of time variation of the gain.

REFERENCES

1. V. M. Popov, "Absolute Stability of Nonlinear Systems of Automatic Control," Automation and Remote Control, pp. 857-875, March, 1962.

2. Kumpati S. Narendra and R. M. Goldwyn, "A Geometrical Criterion for the Stability of Certain Nonlinear Nonautonomous Systems," IEEE Trans. on Circuit Theory, Vol. CT-11, pp. 406-408, Sept., 1964.

3.(a) I. W. Sandberg, "Some Results on the Theory of Physical Systems Governed by Nonlinear Functional Equations," The BSTJ, Vol. XLIV, No. 5, pp. 871-898, May-June, 1965.

(b) I. W. Sandberg, "A Frequency Domain Condition for the Stability of Feedback Systems Containing a Single Time-Varying Nonlinear Element," The BSTJ, Vol. XLIII, No. 4, pp. 1601-1608, July, 1964.

(c) I. W. Sandberg, "On Generalizations and Extensions of the Popov Criterion," IEEE Trans. on Circuit Theory, pp. 117-118, March, 1966.

4.(a) G. Zames, "Nonlinear Time-Varying Feedback Systems—Conditions for L^∞-Boundedness Derived Using Conic Operators on Exponentially Weighted Spaces," Proc. 1965 Allerton Conf., pp. 460-471.

(b) G. Zames, "On the Input-Output Stability of Time-Varying Nonlinear Feedback Systems," Pts. I and II, IEEE Trans. on Automatic Control, Vol. AC-11, pp. 228-238, April and pp. 465-476, July, 1966.

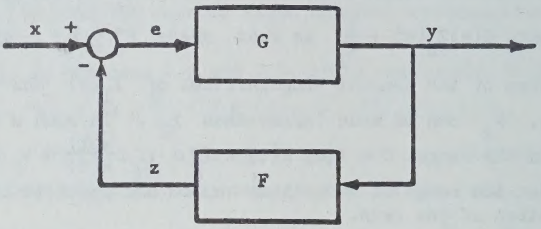
5. R. W. Brockett and L. Forys, "On the Stability of Systems Containing a Time-Varying Gain," Proc. 1964 Allerton Conf., pp. 413-430.

6. C. T. Lee and C. A. Desor, "Stability of Single-Loop Nonlinear Feedback Systems," Elec. Res. Lab., Univ. of Cal., Berkeley, Cal., Rpt. No. ERL 66-13, May, 1966.

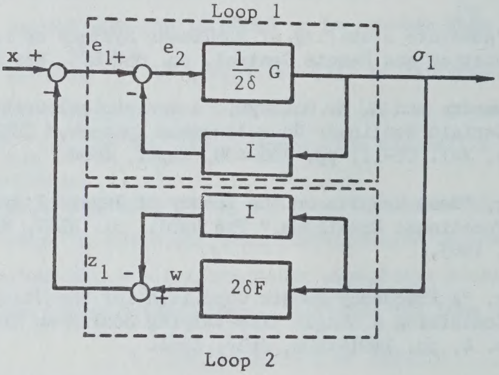
7. Y. S. Cho and Kumpati S. Narendra, "A Circle Criterion for the Stability of Feedback Systems with a Monotonic Nonlinearity," Dunham Lab., Yale Univ., New Haven, Conn., Tech. Rpt. CT-7, Nov., 1966.

8.(a) Kumpati S. Narendra and Y. S. Cho, "Stability of Linear Time-Varying Feed-Systems," Dunham La., Yale Univ., New Haven, Conn., Tech. Rpt. CT-8, Feb., 1967.

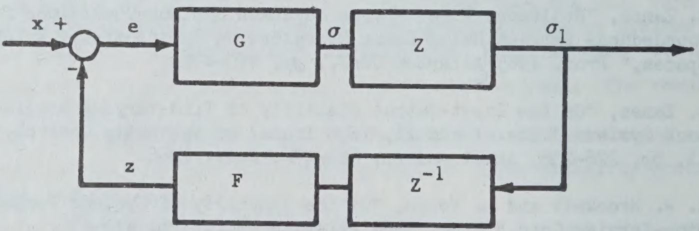
(b) Kumpati S. Narendra and Y. S. Cho, "Stability of Feedback Systems Containing a Single Odd Monotonic Nonlinearity," IEEE Trans. on Automatic Control, Vol. AC-12, Aug., 1967.



(a) Nonlinear Time-Varying Feedback System


Loop 1
x +
e_1+
e_2
$\frac{1}{2\delta}$ G
σ_1
-
-
I
I
z_1 -
w
2δF
+
Loop 2

(b) Proof of Theorem 2.1


x +
e
G
σ
Z
σ_1
-
z
F
Z^{-1}

(c) Proof of Theorem 3.1

Fig. 1

THE PROBLEM OF LUR'E IN SYSTEMS WITH TIME LAG

G.S. TAHIM
Electrical Engineering Department
University of Kansas
Lawrence, Kansas

ABSTRACT

The asymptotic stability of a linear infinite dimensional system with a nonlinear controller in the feedback path is studied. It is known that the hypothesis of complete controllability of the linear part of the system is not needed to assure the existance of a positive definite Liapunov function in the case of a linear finite dimensional system. The same result is seen to be a little easier to establish in the case of an infinite dimensional system.

INTRODUCTION

The problem of the asymptotic stability of an "indirect control" has been studied by many authors recently [1-7]. Lur'e [1] initiated this study and Popov [2], Kalman [4] and Meyer [5,7] established significant results in its solution. Popov obtained a criterion for asymptotic stability of the indirect system without the use of Liapunov functions. It was shown by Meyers [5] that the Popov criterion implies the existance of a suitable positive definite Liapunov function such that its derivative along the trajectories of the system is negative definite.

Yacubovich [6] and Kalman [4] proved the existance of a suitable Liapunov function if the linear part of the system satisfies certain restrictions. The hypothesis of complete controllability and complete observability were needed by Kalman in the proof of his major result. Meyer [7] recently proved that the main result of Kalman is still valid if the hypothesis of complete controllability and complete observability is dropped.

In this note, the asymptotic stability of the indirect control of a system with delay is studied.

In the rest of the paper, upper case letters denote matrices, lower case letters stand for vectors and Greek letters denote scalars. Transpose of a matrix is denoted by a prime. $x(\cdot)$ represents the segment of trajectory over one delay interval.

STATEMENT OF THE PROBLEM

Consider an n^{th} order linear differential-difference system with a nonlinear controller in the feedback path:

$$\begin{aligned} \dot{x}(t) &= A\,x(t) + B\,x(t-1) - c\varphi(\sigma) \\ \dot{\xi}(t) &= -\varphi(\sigma) \\ \sigma &= h'\,x(t) + \rho\xi(t) \end{aligned} \qquad (1)$$

$\varphi(\sigma)$ is a real valued continuous function which belongs to the class F_K such that (1) $\varphi(0) = 0$ and (ii) $0 < \sigma\varphi(\sigma) < \sigma^2 K$. The Lur'e problem can be stated as: Find conditions on ρ, c, h, A and B which are sufficient for the existance of a Liapunov functional of a special type which assures asymptotic stability in the large (AS1L) for system (1) for any $\varphi \in F_\infty$.

The Liapunov functional we shall consider is:

$$V(x(\cdot),\sigma) = x'(t)P\,x(t) + \beta\int_{t-1}^{t} x'(\tau)\,x(\tau)d\tau + \alpha_1\int_0^{\sigma}\varphi(\sigma)d\sigma +$$

$$+ \alpha_2(\sigma - h'x)^2 \qquad (5)$$

$P > 0$, $\beta > 0$; α_1 and α_2 are non-negative constants.

THE STABILITY RESULTS

Consider an unforced linear differential-difference system:

$$\dot{x}(t) = A\,x(t) + B\,x(t-1) \qquad (6)$$

The Liapunov functional selected to study the stability of this system is [8],

$$V(x(\cdot)) = x'(t)P\,x(t) + \beta\int_{t-1}^{t} x'(\tau)\,x(\tau)d\tau \qquad (7)$$

$P > 0$, $\beta > 0$.

Assume that the system without lag ie with $B \equiv 0$ is stable. This implies that given a positive definite matrix $Q > 0$, there exists a $P > 0$ such that

$$A'P + PA = -Q \qquad (8)$$

The derivative of (7) wrt t, in view of (6), is, if $B'P = PB$,

$$\frac{d}{dt}V(x(\cdot)) = x'(t)\left[A'P + PA + \beta I\right]x(t) + 2x'(t)\,PB\,x(t-1) -$$

$$- \beta x'(t-1)\,x(t-1) \qquad (9)$$

Define $A'P + PA + \beta I = -S$ (10)

If Q is selected as a diagonal matrix in (8), S is diagonal. We assume that S is diagonal in the rest of the paper. (9) can now be rewritten as

$\frac{d}{dt}V(x(\cdot)) \leqq 0$ if in equation (11), we satisfy the following equality

$$2\beta\sqrt{S} = 2PB$$

$$-\beta^2\left[A'P + PA + \beta I\right] = P^2B^2 \qquad (12)$$

If a $\beta > 0$ and a $P > 0$ can be found staisfying equation (12), the stability of (6) is assured.

Now we investigate the stability of (4) with the help of the Liapunov functional (5). The derivative of the functional (5), along the trajectories of system (4), is, if B'P = PB,

$$\frac{d}{dt} V(x(\cdot),\sigma) = x'(t) \left[A'P + PA - \beta I\right] x(t) + x'(t)\, 2PB\, x(t-1) -$$
$$-\beta x'(t-1)\, x(t-1) - 2\varphi(\sigma)\left[Pc - \frac{\alpha_1}{2} A'h - \frac{\alpha_2}{2} \rho h\right]' x(t) -$$
$$\alpha_1 \varphi(\sigma)\, h'B\, x(t-1) - \alpha_1 (h'c + \rho)\, \varphi^2(\sigma) - 2\alpha_2 \sigma \rho \varphi(\sigma) \qquad (13)$$

Define

$$A'P + PA + \beta I = -S$$
$$Pc - \frac{\alpha_1}{2} A'h - \frac{\alpha_2}{2}\rho h = d_o$$
$$h'c + \rho = \gamma \qquad (14)$$

In view of (14), the equation (13) can be written as,

$$-\frac{d}{dt} V(x(\cdot),\sigma) = x'(t) S\, x(t) - x'(t)\, 2PB\, x(t-1) + \beta x'(t-1)\, x(t-1)$$
$$+ 2\varphi(\sigma)\, d_o'\, x(t) + \alpha_1 \gamma \varphi^2(\sigma) + 2\alpha_2 \sigma \rho \varphi(\sigma) - \alpha_1 \varphi(\sigma) h'Bx(t-1) \qquad (15)$$

Now the stability requirements for system (4) can be stated as:

Theorem: The system (4) is ASIL if

(i) $\tau + 2\mathrm{Re}\, h' (jwI - A - Be^{-jw})^{-1} c > 0$, for all real w,

(ii) $\alpha_1 \geq 0, \beta > 0, \alpha_2 \geq 0, \alpha_1 + \alpha_2 > 0$

(iii) $\tau = 0$ or $\tau > 0$, $d_o = 0$, $\alpha_2 > 0$

(iv) $h'B = 0$, $B'P = BP$

(v) $A'P + PA = -Q$, $P > 0$ and $Q > 0$.

Proof: (ii) of theorem implies that $V(x(\cdot),\sigma)$ is positive definite. From (ii), it is assured that the first three terms in (15) are positive definite. Adding and substracting $x'(t)\, qq'x(t)$ from the next four terms gives

$$2\varphi(\sigma) d_o'\, x(t) + \alpha_1 \gamma\, \varphi^2(\sigma) + 2\alpha_2 \sigma \rho \varphi(\sigma) - \alpha_1 \varphi(\sigma)\, h'Bx(t-1)$$
$$+ x'(t) qq'x(t) - x'(t)\, qq'\, x(t) = (\sqrt{\alpha_1 \gamma}\, \varphi(\sigma) + q'\, x(t))^2 -$$
$$- \alpha_1 \varphi(\sigma)\, h'B\, x(t-1) + 2\alpha_2 \rho \varphi(\sigma) - x'(t)\, qq'\, x(t) \quad (16)$$

In view of (16), we can rewrite (15) as

$$-\dot{V}(x(\cdot),\sigma) = x'(t) S\, x(t) - x'(t)\, 2PB\, x(t-1) + \beta x'(t-1)\, x(t-1)$$
$$+ (\sqrt{\alpha_1 \gamma}\, \varphi(\sigma) + q'\, x(t))^2 - x'(t)\, qq'\, x(t) - \alpha_1 \varphi(\sigma)\, h'Bx(t-1)$$

$$+ 2\alpha_2\sigma\rho\varphi(\sigma) \qquad (17)$$

Since $h'B = 0$, $\alpha_2 \geqq 0, \sigma > 0$, $-\dot{V}(x(\cdot),\sigma)$ is positive definite and the stability of (4) is assured. The existance of a $P > 0$, $\beta > 0$ implies (i) is satisfied for all real w [4].

CONCLUSIONS

The condition $B'P = BP$, if $P > 0$, implies that B must be diagonal; a diagonal B matrix implies that the linear system (6) is not completely controllable [9].

Thus in the study of asymptotic stability of a linear system, with a nonlinear controller in the feedback path, the hypothesis of complete controllability is not needed, a fact already known but more easily established in this study.

REFERENCES

1. Lurie, A. I., On Some Nonlinear Problems in the Theory of Automatic Control, H. M. Stationery Office, London, 1951.

2. Popov, V. M., Absolute Stability of Nonlinear Systems of Automatic Control, Avt i Telemkh, 22, 1961, pp. 961-979.

3. Yacubovich, V. A., Absolute Stability of Nonlinear Control Systems in the Critical Cases, Avtomat i Telemekh, 24, 1963, pp. 293-303, 717-731.

4. Kalman, R. E., Liapunov Functions for the Problem of Lur'e in Automatic Control, Proc. of National Academy of Sciences, vol. 49, No. 2, 1963, pp. 201-205.

5. Meyer, K. R., Liapunov Functions for the Problem of Lur'e, Proc. of National Academy of Sciences, vol. 53, 1965, pp. 501-503.

6. Yacubovich, V. A., The Solution of Certain Matrix Inequalities in Automatic Control Theory, Dokl. Akad. Nauk. SSSR 143, 1962, pp. 1304-1307.

7. Meyer, K. R., On the Existence of Liapunov Functions for the Problem of Lur'e, JSIAM on Control, Ser. A, vol. 3, No. 3, 1966.

8. Diver, R. D., Existence and Stability of Solutions of a Delay Differential System, Archive for Rational Mechanics and Analysis, vol. 10, No. 5, 1962, pp. 401-426.

9. Tahim, G. S., Optimum Control of Linear Systems with Time Lags Technical Report No. R-242, Coordinated Science Laboratory, University of Illinois, Urbana, Illinois, Jan. 1965.

10. Bellman, R. and Cooke, K. L., Differential-Difference Equations, Academic Press, 1963.

DRIVING POINT IMMITTANCE SYNTHESIS USING EXPONENTIAL LINES

J.C. GIGUERE, M.N.S. SWAMY and B.B. BHATTACHARYYA
Department of Electrical Engineering
Nova Scotia Technical College
Halifax, Canada

ABSTRACT

A synthesis procedure realizing certain types of irrational functions of s as driving point immittances, is presented which uses rc or lossless exponential lines and includes Wyndrum's procedure for the uniform line as a special case. The immittance functions that are realized with exponential rc-lines may be particularly useful in thin film and integrated circuits.

INTRODUCTION

Integrated and thin film circuits are being increasingly used because of their inherent advantages. As a result, the problem of synthesis using distributed rc-lines is receiving some attention. Wyndrum[1] and O'Shea[2] have shown that certain types of irrational functions of s may be realized as driving point immittances by using uniform rc-lines. They have achieved this by introducing the transformations $s_1 = \tanh(as)$ and $s_1 = \cosh\sqrt{a\, s}$, a being a constant, thereby converting the distributed synthesis problem to one of a lumped synthesis in the s_1-plane.

The exponential rc-line $\overline{\text{(ERC)}}$ is supposed to possess certain properties which are either superior to or not obtainable by a uniform line, due to the presence of the taper parameter[1]. Hence a synthesis procedure using exponential lines will make it possible to realize certain types of irrational functions as immittances, which are not realizable by uniform lines. Since the tapering may be varied, it may be observed that the use of $\overline{\text{ERC}}$'s in the synthesis would lead to a better approximation of a given response curve. In this paper, we propose a synthesis procedure using exponential lines.

SYNTHESIS PROCEDURE

Consider the $\overline{\text{ERC}}$ of length ℓ given by

$$r(x) = r_o \exp(2Kx) \; ; \quad c(x) = c_o \exp(-2Kx) \tag{1}$$

r_o and c_o being the resistance and capacitance per unit length and K, the taper parameter which may be positive, negative or zero. For such a line z_{11} and y_{11} are given by[3]

$$Z_{1rc} = z_{11} = (\gamma_o/sc_o) \cosh \gamma_o \ell + (K/sc_o) \tag{1a}$$

and

$$y_{11} = (\gamma_o/r_o) \cosh \gamma_o \ell - (K/r_o) \tag{1b}$$

where

$$\gamma_o = \sqrt{K^2 + sr_oc_o} \,. \tag{1c}$$

Hence for the $\overline{\text{ERC}}$ of length ℓ , given by

$$r(x) = r_o \exp(-2Kx); \quad c(x) = c_o \exp(2Kx) \tag{2a}$$

we have

$$Y_{2rc} = y_{11} = (\gamma_o/r_o) \cosh \gamma_o \ell + (K/r_o). \tag{2b}$$

Thus from (1a) and (2b) we have that z_{11} of line (1) and y_{11} of line (2) are respectively given by

$$Z_{1rc} = \sqrt{\frac{R}{C}}\ (\gamma \coth \gamma + b)\ \frac{1}{s\sqrt{RC}} \tag{3a}$$

and

$$Y_{2rc} = \sqrt{\frac{C}{R}}\ (\gamma \coth \gamma + b)\ \frac{1}{\sqrt{RC}} \tag{3b}$$

where

$$b = K\ell \,, \quad R = r_o\ell \,, \quad C = c_o\ell \tag{4a}$$

and

$$\gamma = \sqrt{b^2 + sRC} \,. \tag{4b}$$

Now making the r - sL transformation[4] we have

$$Z_{1LC} = \sqrt{\frac{R}{C}}\ (\gamma' \coth \gamma' + b)\ \frac{1}{s\sqrt{RC}} \tag{5a}$$

and

$$Y_{2LC} = \sqrt{\frac{C}{R}}\ (\gamma' \coth \gamma' + b)\ \frac{1}{s\sqrt{RC}} \tag{5b}$$

where

$$\gamma' = \sqrt{b^2 + s^2\ RC} \,. \tag{6}$$

It is to be noted that in fact Z_{1LC} and Y_{2LC} correspond respectively to z_{11} and y_{11} of the two exponential LC-lines ($\overline{\text{ELC}}$),

$$L(x) = r_o \exp(2Kx) \;; \quad C(x) = c_o \exp(-2Kx) \tag{7a}$$

and

$$L(x) = r_o \exp(-2Kx) \;; \quad C(x) = c_o \exp(2Kx) \tag{7b}$$

where r_o and c_o are now the inductance and capacitance per unit length of the two $\overline{\text{ELC}}$ s (7a) and (7b).

Now introducing the transformation

$$s_2 = \frac{1}{s_1} = \frac{1}{s\sqrt{RC}}\ (\gamma' \coth \gamma' + b) \tag{8}$$

we have

$$Z_{1LC} = (\sqrt{R/C}\ )\ (1/s_1) = (\sqrt{R/C})\,s_2 \tag{9a}$$

and

$$Z_{2LC} = (1/Y_{2LC}) = (\sqrt{R/C})\ s_1 = (\sqrt{R/C})\ (1/s_2)\ . \tag{9b}$$

Thus, Z_{1LC} and Z_{2LC} correspond to the impedances of a capacitor $C_{s_1} = \sqrt{C/R}$ and an inductor $L_{s_1} = \sqrt{R/C}$ in the s_1-plane. Dual statements may be made about the transformations into the s_2-plane. It may also be observed that the open circuited $\overline{\text{ERC}}$ (1) transforms to a capacitor $C_{s_1} = \sqrt{C/R}$ and that the short circuited $\overline{\text{ERC}}$ (2) transforms to an inductor $L_{s_1} = \sqrt{R/C}$. Again, dual statements may be made about transformations into the s_2-plane. It may be seen that when K = 0, (9) reduces to Wyndrum"s transformations for a uniform line[1].

Hence, given any reactance function in the s_1-plane, say

$$Z_{LC}(s_1) = K_\infty s_1 + \frac{K_o}{s_1} + \Sigma\ \frac{K_i\ s_1}{s_1^2 + \alpha_i^2}\ , \tag{10}$$

it may be synthesized as an LC network in the s_1-plane containing different C_{s_1} and L_{s_1} which may then be replaced respectively by open circuited $\overline{\text{ELC}}$ given by (5a) and the short circuited $\overline{\text{ELC}}$ given by (5b).

We shall now discuss how these results may be made use of in establishing a synthesis procedure for $\overline{\text{ERC}}$'s. Suppose we are given a driving point impedance function which can be expressed in the form

$$Z(s) = \frac{K_\infty}{q(s)} + \frac{K_o\ q(s)}{s} + \Sigma\ \frac{K_i q(s)}{s + q^2(s)\ \alpha_i^2} \tag{11}$$

where

$$q(s) = (\gamma \coth \gamma + b) \tag{12}$$

and γ is given by (4b). Then we perform the r - sL transformation to reduce (11) to the form of (10) in the s_1-plane and synthesize the resulting reactance function in the s_1-plane by a network containing different C_{s_1} and L_{s_1} . Then we replace the different C_{s_1} by open circuited $\overline{\text{ERC}}$'s given by (1) with $C_{s_1} = \sqrt{C/R}$ and the inductance L_{s_1} by short circuited $\overline{\text{ERC}}$'s given by (2) with $L_{s_1} = \sqrt{R/C}$. It is to be noted that the taper factor and the RC product of all these lines are the same as seen from (12) and (4b).

Of course, we could proceed on a dual basis and realize the driving point function as an admittance. Also, by considering the z_{22} of line (1) and y_{22} of line (2), same results may be obtained.

EXAMPLE

To illustrate the procedure consider the realization of the impedance function

$$Z(s) = \frac{2}{q(s)} + \frac{q(s)}{s} + \frac{q(s)}{s + q^2(s)} \qquad (13)$$

where

$$q(s) = \sqrt{b^2 + 4s}\ \coth\left(\sqrt{b^2 + 4s}\right) + b. \qquad (14)$$

Hence from (12) and (4b) we get $K = (b/\ell)$ and $RC = 4$. Assuming unit length for each of the $\overline{\mathrm{ERC}}$, we have $K = b$ and $r_o c_o = 4$. Now the LC impedance function corresponding to (13) is

$$Z_{LC}(s) = \frac{2s}{\gamma' \coth \gamma' + b} + \frac{\gamma' \coth \gamma' + b}{s} + \frac{s(\gamma' \coth \gamma' + b)}{s^2 + (\gamma' \coth \gamma' + b)^2} \qquad (15)$$

where

$$\gamma' = \sqrt{(b^2 + 4s^2)}.$$

Hence, using the transformation (8), namely,

$$s_1 = 2s/(\gamma' \coth \gamma' + b)$$

equation (15) reduces to

$$Z_{LC}(s_1) = s_1 + \frac{2}{s_1} + \frac{2\, s_1}{s_1^2 + 4}.$$

The realization of $Z_{LC}(s_1)$ in the s_1-plane is shown in Fig. 1(a). We may now calculate the parameters of the different lines by using the value of $r_o c_o$ and the fact that $L_{s_1} = \sqrt{(R/C)} = \sqrt{(r_o/c_o)}$ corresponding to the $\overline{\mathrm{ERC}}$ (2) and $C_{s_1} = \sqrt{(c_o/r_o)}$ corresponding to the $\overline{\mathrm{ERC}}$ (1). Then we get the realization of (13) using the $\overline{\mathrm{ERC}}$ lines and is shown in Fig. 1(b). Similarly the other Foster as well as the two Cauer forms may also be obtained. When b = 0, the different lines reduce to uniform lines and the realization is the same as that of Wyndrum[1] for the function

$$Z(s) = \frac{\tanh(2\sqrt{s})}{\sqrt{s}} + \frac{2}{\sqrt{s}\,\tanh(2\sqrt{s})} + \frac{2\tanh(2\sqrt{s})}{\sqrt{s}\,(\tanh^2 2\sqrt{s} + 4)}.$$

CONCLUSIONS

Thus, a synthesis procedure realizing driving point immittances using exponential lines has been presented and the types of irrational functions that may be realized as driving point immittances have been given. The procedure includes as a special case that of Wyndrum for uniform rc-lines.

ACKNOWLEDGEMENTS

This work was supported under the National Research Council of Canada, Grant No. A-2875 and is based on a part of the Ph.D. thesis to be submitted by J.C. Giguere under the supervision of Dr. M.N.S. Swamy, to the Nova Scotia Technical College, Canada.

REFERENCES

1. Wyndrum, R.W., Jr., "The Exact Synthesis of Distributed RC Networks", New York University Tech. Report 400-76, May 1963.
2. O'Shea, R.P., "Synthesis Using Distributed RC Networks", IEEE International Convention Record, Vol. 13, Part 7, pp. 18-29.
3. Ghausi, M.S. and G.J. Herskowitz, "The Effects of Network Shaping and Loading on the Transient and Frequency Response of Distributed RC Networks", New York University Tech. Report 400-71, August 1963.
4. Swamy, M.N.S., V. Ramachandran and B.B. Bhattacharyya, "Lumped Equivalent Circuits of an Exponential rc-Line", to appear in Proc. IEEE, August 1967.

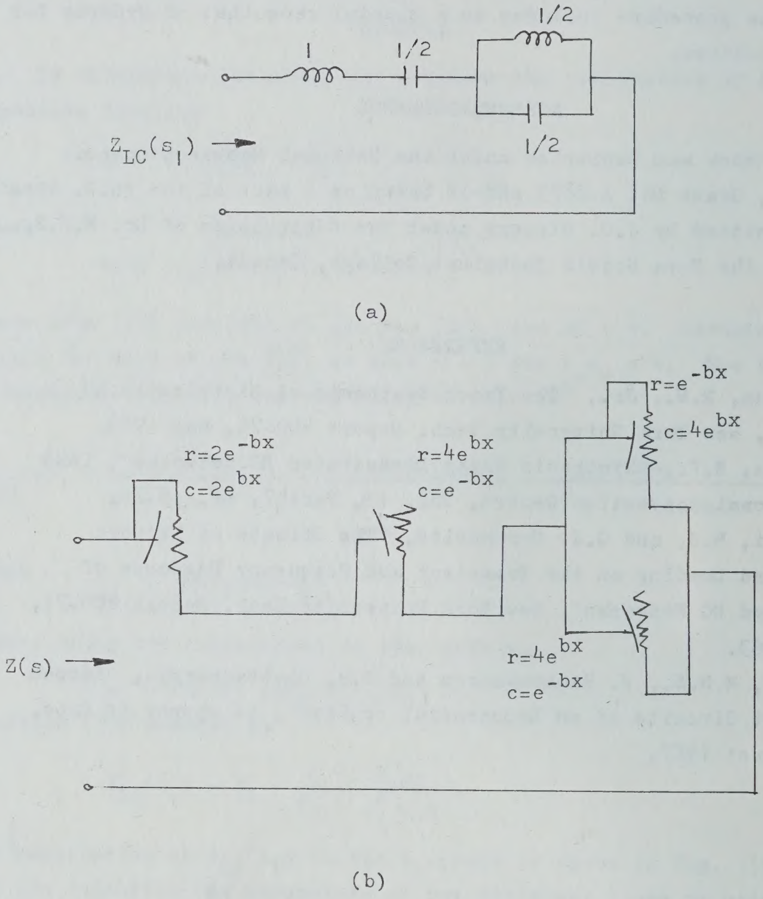

1
1/2
1/2
1/2
$Z_{LC}(s_1)$
(a)
$r=e^{-bx}$
$c=4e^{bx}$
$r=2e^{-bx}$
$c=2e^{bx}$
$r=4e^{bx}$
$c=e^{-bx}$
$r=4e^{bx}$
$c=e^{-bx}$
$Z(s)$
(b)

Figure 1. The synthesis of equation (13) using Foster's first form (a) the s_1-plane structure (b) the actual network realization. All values are in ohms, farads and henries.

THE REALIZABILITY AND DESIGN OF SYMMETRICAL n-PORT TRANSMISSION-LINE NETWORKS

CHARLES V. SHAFFER
Department of Electrical Engineering
University of Florida, Gainesville

ABSTRACT

The necessary and sufficient conditions for the cascade, transformerless realizability of a lossless, symmetrical, transmission-line n-port are given in terms of scalar functions which result from the decomposition of Y(p) or Z(p) ($p = \tanh s/4f_o$) into orthogonal matrix components.

For the 3-port case, a complete design procedure has been developed. This procedure begins with the desired insertion loss characteristics, and yields the line dimensions. The response of a 5th-order Butterworth realization is presented.

INTRODUCTION

Because of the isomorphism pointed out by Richards [1], it is well known that the necessary and sufficient conditions for the realization of an n-port consisting of lossless transmission lines of commensurate lengths, resistors, ideal transformers, and ideal gyrators follow from the realizability conditions for lossless lumped n-ports [2]. However, a practical transmission-line network should not require either ideal or conventional transformers, and the reciprocal networks should not require gyrators. Therefore, we have sought the more restrictive conditions under which a reciprocal transmission-line n-port can be realized without transformers or gyrators [3].

We have concentrated upon a particularly useful canonical transformerless realization, suggested for 2-ports by Saito [4], which consists of a cascade of (n+1)-conductor uniform transmission lines of commensurate lengths terminated in open- and short-circuits, as shown in Figure 1.

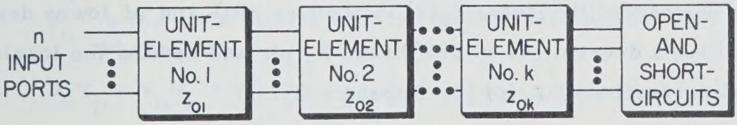


FIG. 1. CANONICAL FORM OF SYNTHESIS

The basic building block is the 2n-port "unit element" shown in Figure 2. (A unit element is one quarter wavelength long at f_o Hertz.)

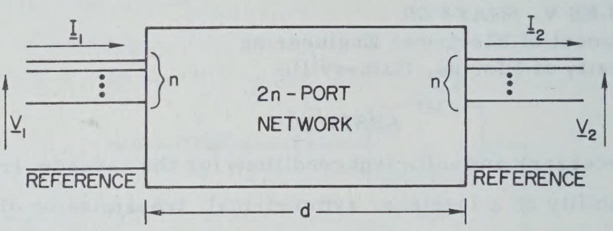


FIG. 2. THE (n+1)-CONDUCTOR (2n-PORT) UNIT ELEMENT

If the "right" n-ports of a unit element are terminated by an n-port with impedance $Z_2(p)$, then the driving-point impedance at the "left" n-ports is given by

$$Z_1(p) = [Z_2(p) + pZ_o]\,[1_n + pZ_o^{-1}Z_2(p)]^{-1} \qquad (1)$$

where 1_n is the n x n identity matrix, $Z_o = Z_1(1)$ is the n x n constant "characteristic impedance matrix" of the unit element, and we temporarily assume that all matrices exist. (The Z's may be replaced by Y's for the dual result. When neither Z_2 nor $Y_2 = Z_2^{-1}$ exists, see [3, p. 14].)

For the cascade synthesis, we need to solve (1) for $Z_2(p)$.

$$Z_2(p) = Z_1(1)\,[Z_1(1) - pZ_1(p)]^{-1}[Z_1(p) - pZ_1(1)] \qquad (2)$$

which is a matrix form of Richards' theorem [5;6]. The formal synthesis procedure therefore begins with a matrix, $Z_1(p)$, which satisfies the well-known realizability conditions for an impedance matrix of a reciprocal, lossless network lumped in p. Principally, $Z_1(p)$ is rational, symmetric, and has, as singularities, only simple poles on the Im(p) axis with positive semi-definite residue matrices; the functions on the main diagonal, $z_{ii}(p)$, everywhere satisfy $z_{ii}(p) = -z_{ii}(-p)$. $Z_1(1) = Z_{o1}$ is the characteristic impedance matrix of the first unit element and $Z_2(p)$ is the impedance "looking into" the second unit element.

By Richards' theorem, $Z_2(p)$ is positive real and of lower degree in p (McMillan's degree, [7, p. 580]) than $Z_1(p)$, and retains the lossless, reciprocal properties of $Z_1(p)$ [3, Appendix C].

Iterating this procedure, we obtain the characteristic impedance matrix of each unit element and the character of the lossless final terminations. In the most general case, it may appear impossible to complete the synthesis because one of the characteristic impedance matrices is non-realizable or because the remainder impedance matrix, $Z_i(p)$, is singular for all p. Reference [3] reviews the strategies to follow in the latter event (pp. 37-40; 45-47). It is sometimes necessary to use the admittance version of (2). Here we wish only to consider the special class of n-ports which we call "symmetrical", exhibiting the realizability constraints on the characteristic immittance matrices (including singular ones) and the resulting necessary and sufficient conditions on $Z_1(p)$ imposed by the chosen canonical form.

ORTHOGONAL DECOMPOSITION OF THE Z MATRIX OF A SYMMETRICAL n-PORT

Definition: A symmetrical n-port is a network having the property that each port is indistinguishable from another by means of external measurements of the port variables (e. g. voltage and current, or incident and reflected waves). It follows that the scattering and immittance matrices of a symmetrical n-port must be of the form [8]

$$\begin{bmatrix} f & g & g & \cdot & \cdot & \cdot & g \\ g & f & g & \cdot & \cdot & \cdot & g \\ g & g & f & \cdot & \cdot & \cdot & g \\ \cdot & \cdot & \cdot & \cdot & & & \cdot \\ \cdot & \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & & \cdot & \cdot \\ g & g & g & \cdot & \cdot & \cdot & f \end{bmatrix} \tag{3}$$

Theorem 1: The open-circuit impedance (short-circuit admittance, scattering) matrix of a symmetrical n-port (when it exists) always admits the decomposition

$$Z(p) = z_u(p)\, U_n + z_v(p)\, V_n \tag{4}$$

where: p may be either the complex frequency or some function of it;

z_u and z_v are scalar functions;

U_n is the n x n matrix of all 1's;

V_n is the n x n matrix with n-1 on the main diagonal and -1 elsewhere; and

$$U_n V_n = V_n U_n = 0_n \tag{5}$$

The proof follows directly from the definition of a symmetrical n-port. Specifically, using the notation of (3) for Z(p),

$$z_u = \frac{f + (n-1)g}{n}; \qquad z_v = \frac{f-g}{n} \tag{6}$$

An example follows.

$$\frac{1}{p}\begin{bmatrix} p^2+2 & 1 & 1 \\ 1 & p^2+2 & 1 \\ 1 & 1 & p^2+2 \end{bmatrix} = \frac{p^2+4}{3p}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} + \frac{p^2+1}{3p}\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

$$= \frac{p^2+4}{3p} U_3 + \frac{p^2+1}{3p} V_3$$

A stronger result is available, as follows:

Theorem 2: If an open-circuit impedance or short-circuit admittance matrix of a (linear, constant) passive network, lumped in p, can be represented by (4), then $z_u(p)$ and $z_v(p)$ are positive-real functions; if the network is lossless, they are lossless immittance functions.

For an indirect proof, see [3, p. 55].

Because U_n and V_n are mutually orthogonal, it follows that matrix operations on port descriptions of symmetrical n-ports can be accomplished by (much simpler) scalar operations. Specifically, our cascade synthesis benefits from the following theorem which obviates the use of the matrix version of Richards' theorem.

Theorem 3: Given the n x n rational matrix (already defined)

$$Z_1(p) = z_{u1}(p)\, U_n + z_{v1}(p)\, V_n \tag{7}$$

and the matrix Richards' transformation

$$Z_2(p) = Z_1(1)\,[Z_1(1) - pZ(p)]^{-1}[Z_1(p) - pZ_1(1)] \tag{2}$$

then $Z_2(p)$ is given by

$$Z_2(p) = z_{u2}(p)\, U_n + z_{v2}(p)\, V_n \tag{8}$$

where the latter scalar functions, z_{u2} and z_{v2}, are related to the former by the scalar Richards' transformation

$$z_{u2}(p) = z_{u1}(1)\,\frac{z_{u1}(p) - pz_{u1}(1)}{z_{u1}(1) - pz_{u1}(p)} \tag{9a}$$

$$z_{v2}(p) = z_{v1}(1)\,\frac{z_{v1}(p) - pz_{v1}(1)}{z_{v1}(1) - pz_{v1}(p)} \tag{9b}$$

The proof may be carried out by substituting (7) in (2), using (5) and the following additional properties of U_n and V_n [3, Appendix B]

$$\text{Inversion:} \quad (aU_n + bV_n)^{-1} = \frac{1}{n^2 a} U_n + \frac{1}{n^2 b} V_n \tag{10}$$

$$\text{Squaring:} \quad U_n^2 = nU_n ; \qquad V_n^2 = nV_n \tag{11}$$

Note from (9b) that if $z_{v1}(p)$ has a zero at $p = 0$, then $z_{v2}(p)$ also has a zero at $p = 0$ (if, in addition, $z_{v1}(p)$ is only of first degree, $z_{v2}(p)$ is identically zero for all p). The poles of $z_{v1}(p)$ at $p = 0$ are also preserved by (9b). Similar remarks apply to (9a).

NECESSARY AND SUFFICIENT CONDITIONS FOR SYMMETRICAL 3-PORT CASCADE REALIZATION

Because in our canonical form, physically symmetrical [8] n-ports do not exist for n>3, we now consider 3-ports specifically. We know that each (6-port) unit element must have a hyperdominant characteristic admittance matrix [9], if it exists. Also, we find that, because of the required symmetry, if Z_{oi} is singular, it must be $z_{oi}U_3$, and if Y_{oi} is singular, it must be $y_{oi}V_3$ (or, trivially, 0_3 in each case). Which type of singularity we may encounter depends on the relative degrees [7] of the functions z_u and z_v. Moreover, because (9) can't be applied to 1/p, the zero-frequency behavior of z_u and z_v determine whether the first singular extraction is carried out with the impedance or the admittance version of (9). Table I summarizes all 12 possible conditions and classifies the 5 realizable Filter Types.

TABLE I. FILTER TYPES ACCORDING TO PROPERTIES OF Z(p)

at p=0		Case 1	Case 2	Case 3
z_u has	z_v has	$\delta(z_u) > \delta(z_v)$	$\delta(z_u) < \delta(z_v)$	$\delta(z_u) = \delta(z_v)$
Zero	Zero	Type I	Not Allowed	Type III-S
Pole	Zero	Type I	Type II	Type III-H
Pole	Pole	Not Allowed	Type II	Type III-O
Zero	Pole	Not Allowed	Not Allowed	Not Allowed

Each of the allowed Filter Types consists of a cascade of $\hat{k}$ regular unit elements where $\hat{k} = \min\{\delta(z_u), \delta(z_v)\}$, followed by a cascade of m "irregular" stages where $m = |\delta(z_u) - \delta(z_v)|$, and lossless terminations. Specifically, the types are distinguished as follows:

Type I: All three inner conductors, after the $\hat{k}^{th}$ regular unit element, coalesce into one in the remaining cascade of m coaxial line sections.

Type II: The $\hat{k}^{th}$ regular unit element is connected to a WYE network, each leg of which is a cascade of m coaxial line sections.

Type III-S: The $\hat{k}^{th}$ regular unit element is terminated by connecting all 3 inner conductors to the sheath.

Type III-H: The $\hat{k}^{th}$ regular unit element is terminated by connecting only the 3 inner conductors together.

Type III-O: The $\hat{k}^{th}$ regular unit element is not terminated.

Table II supplies the synthesis details for the 3 possible cases.

TABLE II. FILTER TYPES ACCORDING TO REMAINDER FUNCTIONS AFTER $\hat{k}$ (OR $\hat{k}+1$) EXTRACTIONS

Case 1: $\delta(z_u) > \delta(z_v)$		Case 2: $\delta(z_u) < \delta(z_v)$	
If, at $p = 0$, z_v has a		If, at $p = 0$, z_u has a	
zero	pole	zero	pole
then $z_{v\hat{k}}(p)$ is		then $z_{u\hat{k}}(p)$ is	
$b \cdot p$	b/p	$a \cdot p$	a/p
& the $\hat{k}^{th}$ extraction leaves		& the $\hat{k}^{th}$ extraction leaves	
$Z_{\hat{k}+1}(p) =$	$Y_{\hat{k}+1}(p) =$	$Z_{\hat{k}+1}(p)$	$Y_{\hat{k}+1}(p) =$
$z_{u(\hat{k}+1)}(p)\ U_3$	$y_{u(\hat{k}+1)}(p)\ U_3$	$z_{v(\hat{k}+1)}(p)\ V_3$	$y_{v(\hat{k}+1)}(p)\ V_3$
which has		which has	
a Type I realization.	no transformerless realization.	no transformerless realization.	a Type II realization.

Case 3: $\delta(z_u) = \delta(z_v)$			
If, at $p = 0$, z_u has a			
zero	pole	pole	zero
and z_v has a			
zero	zero	pole	pole

(continued)

TABLE II, Case 3 (continued)

	then $Z_{\hat{k}}(p)$ is		
$apU_3 + bpV_3$	$\frac{a}{p}U_3 + bpV_3$	$\frac{a}{p}U_3 + \frac{b}{p}V_3$	$apU_3 + \frac{b}{p}V_3$
	which has		
a Type III-S realization.	a Type III-H realization.	a Type III-O realization.	no transformerless realization.

Finally, the necessary and sufficient conditions are concisely stated in Theorem 4, the proof of which follows (by contradiction and construction, respectively [3, pp. 56-58]) from the previously tabulated conditions. Note that Item 4 in the Theorem is an alternate expression of the hyperdominance condition.

Theorem 4: A rational impedance matrix, Z(p), $p = \tanh s/4f_o$, is realizable by a symmetric, lossless, transformerless, 3-port cascade coupled-transmission-line network if and only if:

1. Z(p) permits the decomposition

$$Z(p) = z_u(p)\, U_3 + z_v(p)\, V_3$$

2. The scalar functions, $z_u(p)$ and $z_v(p)$, are lossless impedance functions of p.
3. Let $\delta(z)$ be McMillan's degree of z(p), then:
   a. If $\delta(z_u) > \delta(z_v)$, then z_v has a zero at $p = 0$.
   b. If $\delta(z_v) > \delta(z_u)$, then z_u has a pole at $p = 0$.
   c. If $\delta(z_u) = \delta(z_v)$ and z_u has a zero at $p = 0$ then z_v also has a zero at $p = 0$.
4. Denote by $z_{ui}(p)$ [or $z_{vi}(p)$] the function resulting from i - 1 applications of Richards' theorem to $z_u(p)$ [or $z_v(p)$]. Then $z_{ui}(1)$ is not less than $z_{vi}(1)$ for $i = 1, 2, \ldots, \hat{k}$, where

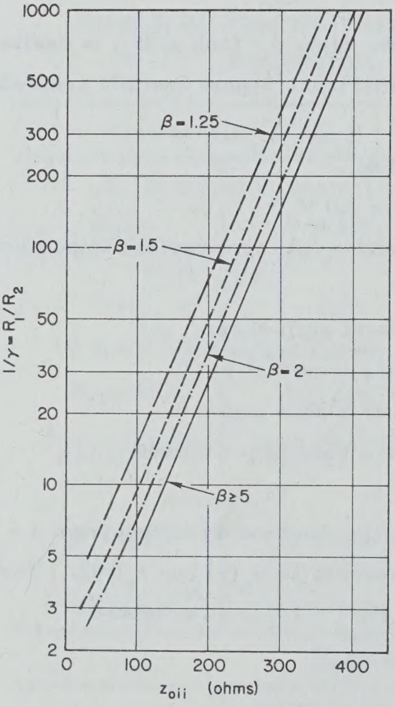
$$\hat{k} = \min\{\delta(z_u), \delta(z_v)\}.$$

FILTER DESIGN SEQUENCE

The complete symmetrical filter design problem consists of the following steps (at each asterisk, a partial realizability check is possible):

1. From given symmetrical specifications, find $Z_1(p)$ using [8].
2. Decompose $Z_1(p)$ by Theorem 1.

3. Find filter classification in Table I. (*)
4. $Z_{o1} = Z_1(1)$ is the characteristic impedance matrix of the first unit element. (*)
5. Apply (9) to find z_{u2} and z_{v2}.
6. $Z_{o2} = Z_2(1)$. (*)
7. Iterate steps 5 and 6 until the synthesis terminates. (*)
8. If all unit elements are realizable, find the approximate line dimensions from Fig. 3 [3, pp. 59-66].
9. Refine dimensions using an electrolytic tank or other potential analog.
10. Construct filter, making each unit element one-quarter wavelength at f_o.



$Z_o = \{z_{oij}\}$

R_1 = I.D. of sheath

R_2 = O.D. of each inner conductor

b = radius of circle on which centers of inner conductors lie

$\beta = R_1/b$

$\gamma = R_2/R_1$

$\beta^2 \cong 3 \exp(z_{oij}/30) - 1$ $(i \neq j)$

FIG. 3. APPROXIMATE RELATION OF CONDUCTOR SIZES TO Z_o FOR A 6-PORT UNIT ELEMENT WITH AIR DIELECTRIC

EXPERIMENTAL RESULTS

Two realizations (Types I and II) of the 3-port coupled-line filter with 5th-order Butterworth insertion loss characteristics have been constructed and tested. Figure 5 pictures the internal assembles of the filters beside the sheath, which was used interchangeably for both of them. Note that the Type II filter is only two-thirds as long as the Type I because of the former's re-entrant final section.

Figure 4 compares the measured insertion loss of the Type I filter with the ideal characteristic. The Type II filter is less satisfactory because of parasitic lumped capacitance effects inherent in its termination [3, pp. 72-73].

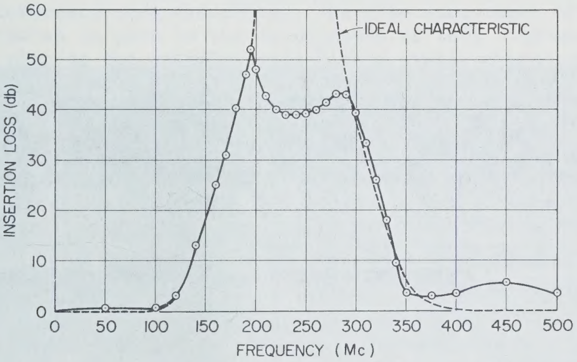


FIG. 4. MEASURED INSERTION LOSS OF THE TYPE I FILTER

REFERENCES

1. Richards, P. I., Resistor-Transmission-Line Circuits, Proceedings of the IRE, vol. 36, no. 2, February, 1948, pp. 217-220.

2. Newcomb, R. W., Nonreciprocal Transmission-Line n-Port Synthesis, Stanford Electronics Laboratories, Technical Report No. 2254-3, September, 1962.

3. Shaffer, C. V., Transformerless n-Port Symmetrical-Transmission-Line Synthesis, Stanford Electronics Laboratories, Technical Report No. 6558-2, August, 1965.

4. Saito, N., A Coupled Transmission Line Filter, The Journal of the Institute of Electrical Communication Engineers of Japan, vol. 44, no. 7, July, 1961, pp. 1036-1040.

5. Richards, P. I., A Special Class of Functions with Positive Real Part in a Half-Plane, Duke Mathematical Journal, vol. 14, no. 3, September, 1947, pp. 777-786.

6. Newcomb, R. W., Richards' Theorem for Matrices, Stanford Electronics Laboratories, Technical Report No. 2254-1, July, 1962.

7. McMillan, B., Introduction to Formal Realizability Theory—II, Bell System Technical Journal, vol. 31, May, 1952, pp. 541-600.

8. Shaffer, C. V., Derivation of n-Ports with Prescribed Insertion-Loss Characteristics from Known 2-Ports, Electronics Letters, vol. 1, no. 7, 1965, pp. 206-207.

9. Youla, D. C., An Introduction to Coupled-Line Theory—Part I, Memorandum 54, PIBMRI-960-61, Polytechnic Institute of Brooklyn, 1961.

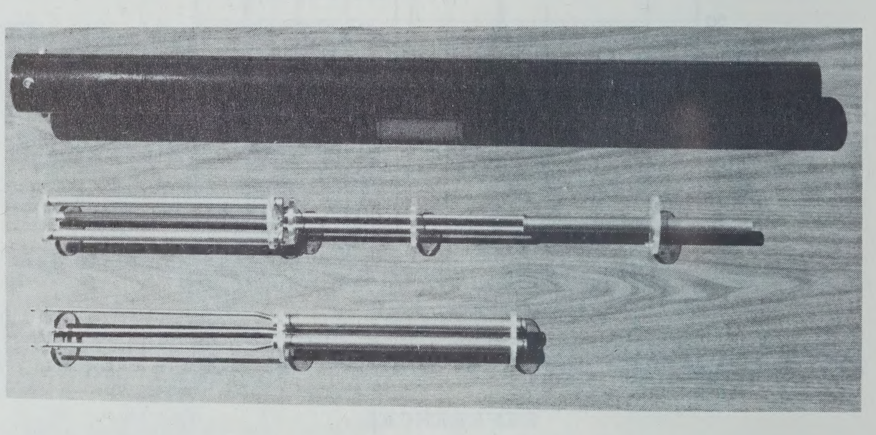


FIG. 5. COMPONENTS OF THE 5th-ORDER BUTTERWORTH 3-PORT FILTERS

Below the sheath are the Type I and Type II internal assemblies. (White rectangle under sheath is a temporary support.)

A GENERAL APPROXIMATION SYNTHESIS METHOD USING UNIFORM $\overline{\text{RC}}$ AND $\overline{\text{RCG}}$ LINES

SULEYMAN S. PENBECI AND SAMUEL C. LEE
New York University
Department of Electrical Engineering
Bronx, New York 10453

ABSTRACT

In this paper, a general approximation synthesis method using uniform $\overline{\text{RC}}$ and $\overline{\text{RCG}}$ lines by means of orthonormal functions (such as Legendre polynomials, Jacobi polynomials, Laguerre polynomials, etc.) is presented which can be used to realize any rational and irrational network functions. The approximation is accomplished in a transformed complex plane where finite $\overline{\text{RC}}$ lines can be represented as R and C circuit elements. Several examples of driving point and transfer function realizations are given.

I. INTRODUCTION

Thin film technology is rapidly becoming one of the basic building blocks of microelectronic circuitry. The wide acceptance of film circuitry can be attributed in part to the development of thin film materials; such as tantalum nitride for example, with highly stable characteristics and the ability to produce such films in quantity with uniform quality. Because of these new developments, network synthesis, using distributed components such as $\overline{\text{RC}}$ or $\overline{\text{RCG}}$ lines, becomes more and more attractive from engineering point of view. This is because thin film distributed networks, in many cases, not only can replace larger and more cumbersome lumped circuit components but also inherently provide interwiring in a small package as a part of the manufacturing process. Although the analysis and synthesis of $\overline{\text{RC}}$ networks has been investigated in the past few years by a number of authors,[1-9] there are mainly two basic approaches to the problem. The first technique involves truncating infinite series expansion of transcendental expressions describing infinite pole-zero pattern of $\overline{\text{RC}}$ lines. The second approach utilizes a complex frequency transformation which provides rational functions to describe $\overline{\text{RC}}$ networks. This method first introduced by Wyndrum[4] transforms the uniform open- and short-circuit finite $\overline{\text{RC}}$ immittance functions ($\overline{\text{URCO}}$ and $\overline{\text{URCS}}$, respectively) into lumped LC immittances in s_1 domain by means of the transformation $s_1 = \tanh\sqrt{rcs}$ or $s_1 = \coth\sqrt{rcs}$. Later O'Shea[6] gave a synthesis procedure using $s_1 = \cosh\sqrt{rcs}$. Recently Stein, Mulligan and Shamis[7] introduced a transfer function synthesis method by transforming $\overline{\text{URC}}$ immittance functions into lumped RC network functions. Transformations which are similar to those just described are used in this paper.

It is well-known that the applicability to the exact synthesis of distributed $\overline{\text{RC}}$ network are limited to a very restrictive class of functions. However, if one uses an approximation realization technique instead of the exact synthesis, the constraints on network function can be removed. In this paper, a general approximation synthesis method using uniform $\overline{\text{RC}}$ and RCG lines by means of orthogonal functions is presented which can be used to realize any rational and irrational network functions.

Historically, the use of orthogonal functions for obtaining approximations is quite old. The approximating properties of Fourier functions in time domain are well-known. The first application of Laguerre's functions to transient problems was due to Y. W. Lee[10] in 1932. Later other orthogonal sets such as Legendre, Jacobi polynomials for time domain approximations were introduced by Papoulis[11] and Armstrong.[12] Besides the time approximations, Y. W. Lee[13] introduced a method obtaining orthogonalization in frequency domain. In all of the work cited above rational and/or irrational functions are being approximated by rational functions. In this paper it is shown that these techniques can also be extended to make approximation in terms of irrational functions of complex frequency. This generalization may result much better convergence for some class of functions.

II. APPROXIMATION THROUGH THE MINIMIZATION OF INTEGRAL SQUARED ERROR

The main consideration in any approximation technique is to make specified error (i.e., difference between desired function and its approximation) within specified tolerances. In this paper, the approximation technique is used to minimize integral squared error defined as:

$$\mathrm{ISE} = \int_0^\infty \left[f_d(v) - f_a(v) \right]^2 dv \tag{1}$$

where $f_d(v)$ is the desired network function $f_a(v)$ is its approximation. The advantage of this method over the other methods (such as curve fitting method etc.) is that when $f_d(v)$ is expanded in terms of orthonormal functions[15] the truncation of this expansion provides optimum approximation provided that $f_d(v)$ is real, continuous almost everywhere and causal (i.e., $f_d(v) = 0$ for $v < 0$).

The following orthonormal polynomials are considered: a) Legendre Polynomials[11]

$$P_n(v) = \frac{1}{2^n n!} \frac{d^n}{dv^n} (v^2-1)^n \tag{2}$$

Changing the variable $v = e^{-\alpha x}$ we define modified Legendre Polynomials

$$g_n(x) = P_{2n}(e^{-\alpha x}) \tag{3}$$

b) Laguerre Polynomials[11]

$$L_n(v) = e^v \frac{d^n}{dv^n} \frac{v^n}{n!} e^{-v} \tag{4}$$

which can be put in modified form by

$$g_n(x) = (-1)^n e^{-\alpha x} L_n(2\alpha x) \tag{5}$$

c) Jacobi Polynomials[12]

$$J_n(p,q,v) = \frac{v^{1-q}(1-v)^{q-p}}{q(q+1)\ldots(q+n-1)} \frac{d^n}{dv^n} v^{q+n-1} \tag{6}$$

and modified Jacobi Polynomials are defined as

$$g_n(x) = (-1)^n \sqrt{2\alpha(n+1)^3}\, e^{-\alpha x} J_n(2,2/e^{-\alpha x}) \tag{7}$$

One can show[11] that in frequency domain the above orthogonal functions will have the following forms: a) Modified Legendre Polynomials:

$$G_o(s) = \frac{1}{s} \quad , \quad G_n(s) = \frac{(s-\alpha)(s-3\alpha)\ldots[s-(2n-1)\alpha]}{s(s+2\alpha)(s+4\alpha)\ldots(s+2n\alpha)} \quad , \quad n=1,2,\ldots \tag{8}$$

b) Modified Laguerre Polynomials:

$$G_n(s) = \frac{s^n}{(s+\alpha)^{n+1}} \quad n=0,1,2,\ldots \tag{9}$$

c) Modified Jacobi Polynomials:

$$G_n(s) = (-1)^n \sqrt{2\alpha(n+1)^3} \sum_{m=o}^{n} \frac{C_{mn}}{s+(m+1)} \qquad \begin{matrix} m=1,2,\ldots,n \\ n=0,1,2,\ldots \end{matrix} \tag{10}$$

where $C_{mn} = (-1)^m \dfrac{(m+n+1)!\ n!}{m!(n-m)!(m+1)!(n+1)!}$

It can be shown [13] that the transforms of orthogonal functions are again orthogonal. The coefficients C_n of expansion F′s) in terms of Laplace transform of orthogonal functions, i.e.,

$$F(s) = \sum_{n=o}^{\infty} C_n G_n(s) \tag{11}$$

are: a) For modified Legendre Polynomials [11]

$$C_o = F(\alpha) \quad , \quad C_n = \frac{(2n+1)(2n+3)\ldots(4n+1)}{2n(2n-2)(2n-4\ldots 2} F[(2n+1)\alpha]$$

$$- \frac{(2n-1)(2n+1\ldots.(4n-3)(4n+1)}{(2n-2)(2n-4)\ldots 2x2} F[(2n-1)\alpha]$$

$$+ \frac{(2n-3)(2n-1\ldots.(4n-5)(4n+1)}{(2n-4)(2n-6\ldots.\ 2x2x4} F[(2n-3)\alpha]$$

$$- \frac{1x3\ldots.(2n-1)(4n+1)}{2x4\ldots\ldots 2n} F(\alpha) , \quad n=1,2,\ldots \tag{12}$$

b) For modified Laguerre Polynomials

$$C_n = \sum_{i=o}^{n} \binom{n}{i} a_{n-i} \tag{13}$$

where a_n are the coefficients of the Taylor series expansion of F(s) about s=o, i.e., $F(s) = \sum_{n=o}^{\infty} a_n s^n$.

c) For modified Jacobi Polynomials

$$C_n = (-1)^n \sqrt{2\alpha(n+1)^3} \sum_{m=o}^{n} C_{mn} F[(m+1)\alpha] \tag{14}$$

It is interesting to note that the above equations are rational functions of s whose poles and zeros are all real. Thus they can be realized by RC and/or active RC networks.

III. $\overline{\text{RC}}$ AND $\overline{\text{RCG}}$ LINES AS A NETWORK ELEMENT:

The open-circuit impedance matrix of a general line of unit length is given by [4]

$$Z = Z_o \begin{bmatrix} \coth\gamma & \operatorname{csch}\gamma \\ \operatorname{csch}\gamma & \coth\gamma \end{bmatrix} \tag{15}$$

Where Z_o is the characteristic impedance of the line and γ is the propogation constant. For a uniform $\overline{\text{RC}}$ line ($\overline{\text{URC}}$) , $Z_o = \sqrt{\frac{r}{cs}}$ and $\gamma = \sqrt{rcs}$, and for a uniform RCG line ($\overline{\text{URCG}}$),

$$Z_o=\sqrt{\frac{r}{cs+g}} \quad \text{and} \quad \gamma=\sqrt{r(cs+g)} \quad ,$$

where r,c and g are the resistance, capacitance and conductance of the line per unit length.

A. Network Synthesis Using $\overline{\text{URCS}}$ $\overline{\text{URCO}}$ Lines:

From (15), the driving point impedance of a line when the second port is open-and short-circuited are found to be

$$Z_{ins} = Z_o \tanh\gamma \tag{16a}$$

$$Z_{ino} = Z_o \coth\gamma \tag{16b}$$

To be able to apply RC (or RL) network synthesis methods to $\overline{\text{URC}}$ network synthesis the following transformations[7] will be used.

Step 1: Pre-multiply the given network function F(s) and define $F_1(s)$ as:

$$\begin{aligned} F_1(s) &= \sqrt{s}\,F(s) \quad \text{for } F(s) \text{ is an impedance function} \\ F_1(s) &= F(s)/\sqrt{s} \quad \text{for } F(s) \text{ is an admittance function} \end{aligned} \tag{17}$$

It can be shown [3] that if F(s) is a network function of an RC network than $F_1(s)$ is a network function of an LC network in the $\sqrt{s}$ plane.

Step 2: Change the variable s to s_1 by the relation

$$s_1 = \tanh\ b\sqrt{s} \tag{18}$$

where b^2 is the rc product of the $\overline{\text{URC}}$ segments and obtain $F_1(s_1)$. One can show that $F_1(s_1)$ is a network function of a lumped LC network.

Step 3: Use LC to RC transformation for impedance function and LC to RL transformation for admittance function.

$$F_2(s_2) = \frac{1}{s_1} F_1(s_1) \Bigg|_{s_1^2 = s_2} \tag{19}$$

It can be proven that a lumped R and C circuit elements (L and 1/R for admittance function) of $F_2(s_2)$ in the s_2 plane correspond to short and open-circuited URC lines ($\overline{\text{URCS}}$ and $\overline{\text{URCO}}$ respectively) in s plane which are shown in Figures 1a. and 1b.

The realization of F(s) using $\overline{\text{URC}}$ lines may be obtained from the realization of $F_2(s_2)$ in the s_2-plane by replacing lumped elements by $\overline{\text{URCS}}$ and URCO lines. However, in general, $F_2(s_2)$ is a very complicated function which by no means can be realized by a lumped RC network.

One can also use s_1=coth $b\sqrt{s}$ transformation in Step 2. of above procedure. In this case lumped R and C circuit elements (L and 1/R for admittance function) of $F_2(s_2)$ in the s_2-plane correspond $\overline{\text{URCO}}$ and $\overline{\text{URCS}}$ lines, respectively, in the s-plane.

B) Network Synthesis Using O'Shea's Synthesis Technique:

The impedance matrix (15) of general line will have the following form for $\overline{\text{RC}}$ lines.

$$Z = \frac{1}{b\sqrt{s}\ \sinh\ b\sqrt{s}} \begin{bmatrix} rs_1 & r \\ r & rs_1 \end{bmatrix} \tag{20}$$

where s_1= cosh $b\sqrt{s}$ and b is defined as before. Recently O'Shea[6] has presented a synthesis technique using complex frequency transformation s_1=cosh $b\sqrt{s}$. For instance, a driving point admittance function which is realizable by this method will have the following form.

$$Y(s) = b\sqrt{s}\ \sinh b\sqrt{s} \sum_{n=o}^{N} \frac{C_n}{s_1+\beta_n} \tag{21}$$

where $-1<\beta_n<1$; $C_n>o$. Each term in the above summation can be realized as an element shown in Figures 1c, and 1d. For $\beta_n>o$, the realization will be in the form shown in Figure 1c, where

$$r_1 = \frac{1-\beta_n}{C_n} \text{ and } r_2 = \frac{2\beta_n}{Cn} \quad .$$

For $\beta_n<o$, it will take the form shown in Figure 1d, where

$$r_3 = \frac{1+\beta_n}{C_n} \text{ and } r_4 = \frac{\beta_n}{2C_n} \quad .$$

Since orthogonal functions given in section II have their poles on the negative real axis, (see (8)(9) and (10)) in order to present the form of Y(s) in (21) in terms of those orthogonal functions, another transformation $s_2 = s_1-1$ is necessary. The procedure will have the following steps.

Step 1: Pre-multiply given admittance function and define $F_1(s)$ as:

$$F_1(s) = \frac{1}{b\sqrt{s}\ \sinh b\sqrt{s}} F(s) \tag{22}$$

Step 2: Change the variable s to s_1 by the relation:

$$s_1 = \cosh b\sqrt{s} \tag{23}$$

and obtain $F_1(s_1)$

Step 3: Change the variable s_1 to s_2 by the relation:

$$s_2 = s_1-1 \tag{24}$$

and obtain $F_2(s_2)$. It can be shown [6] that $F_2(s_2)$ is positive real.

Step 4: After expansion of $F_2(s_2)$ in terms of orthogonal functions, transform back to $F_1(s_1)$ by the relation.

$$s_1 = s_2+1 \tag{25}$$

$F_1(s_1)$ can be realized using circuit elements shown in Figures 1c. and 1d.

The transformations of $\overline{\text{RCG}}$ lines will be exactly the same as for $\overline{\text{RC}}$ lines with an additional initial step of frequency translation of $s'=s+\frac{g}{c}$. In Steps 1 and 2, s is replaced with s' for $\overline{\text{RCG}}$ lines.

IV. APPROXIMATION IN TRANSFORMED COMPLEX FREQUENCY S_2 DOMAIN:

If $f(t) = L^{-1}F(s)$ is causal, real and continuous almost everywhere, then it can be shown [4], [6], [14] that $f(v)= L^{-1}F_2(s_s)$ is also causal, real and continuous almost everywhere. Then we can write:

$$f(v) = \sum_{n=o}^{\infty} C_n\, g_n(v)$$

or,

$$F_2(s_2) = \sum_{n=o}^{\infty} C_n G_n(s_2) \tag{26}$$

Truncation of this series for any finite value of n will result an optimum approximation of $F_2(s_2)$. The number of terms needed will depend on the allowable error specified by the problem.

Notice that in calculating coefficients of the expansion of $F_2(s_2)$ of Eq. (26) if the transformation, $s_1 = \tanh b\sqrt{s}$ is used, s_2 must be smaller than unity. For transformation $s_1 = \coth b\sqrt{s}$ it should be greater than unity. For $s_1 = \cosh b\sqrt{s}$, s_2 is limited between zero and two. Truncation of (26) for this reason will introduce some error, which can vary depending on the type of function and also choice of scaling factor α. Therefore a suitable choice of α is necessary to minimize the error.

V. NETWORK SYNTHESIS USING $\overline{RC}$ AND $\overline{RCG}$ LINES

The approximation of $F_2(s_2)$ can be written in the following form

$$\tilde{F}_2(s_2) = \left[\sum_{i=0}^{N^+} \frac{A_i^+}{s_2+a_i}\right] - \left[\sum_{j=0}^{N^-} \frac{A_j^-}{s_2+a_j}\right] = [\tilde{F}^+(s_2)] - [\tilde{F}^-(s)] \qquad (27)$$

where A_i^+, A_j^-, a_i and a_j are non-negative.

A. Driving Point Immittance Function Synthesis

If given function a driving point impedance function, then $\tilde{F}_2^+(s_2)$ and $\tilde{F}_2^-(s_2)$ can be realized by any of the known RC driving point synthesis methods. The realization of $F_2(s_2)$ is shown in Figure 2(a).

If given function F(s) is a driving point admittance function than $F_2^+(s_2)$ and $F_2^-(s_2)$ can be realized by any of the known RC one-port synthesis as shown in Figure 2(b). The approximation synthesis is illustrated by using modified Legendre polynomials in the following example. Of course, other orthogonal functions can be used as well.

Example of Driving Point Impedance Function Synthesis by Means of Modified Legendre Polynomials

Suppose $Z(s) = 1/s$ is a driving impedance function. $F_1(s)$ is found to be $1/\sqrt{s}$. Then $F_1(s_1) = b/\tanh^{-1}s_1$ for b=1. Therefore, we will have $F_2(s_2) = 1/(\sqrt{s_2}\tanh^{-1}\sqrt{s_2})$. The series expansion of the approximated $F_2(s_2)$ with n=6 and α=.055, using modified Legendre polynomials, will have the following coefficients

$C_0 = 0.9807$ $C_1 = -0.09754$ $C_2 = -0.1982$ $C_3 = -0.3571$

$C_4 = -0.6778$ $C_5 = -1.5444$ $C_6 = -4.899$

The approximation of $F_2(s_2)$ can be written in the following form

$$\tilde{F}_2(s_2) = \left[\frac{0.0713}{s_2} + \frac{71.049}{s_2+.11} + \frac{2229.85}{s_2+.33} + \frac{6240}{s_2+0.55}\right]$$

$$-\left[\frac{1059.46}{s_2+.22} + \frac{9279.6}{s_2+.44} + \frac{3210}{s_2+.66}\right] = [\tilde{F}_2^+(s_2)] + [\tilde{F}_2^-(s_2)] \qquad (28)$$

which can be realized by a network shown in Figure 3. In this example the value of the capacitance of the line is chosen as the reciprocal of the value of the resistance to make b=1. Of course, other values of b can be used in the realization.

As an example of realization of admittance function, we shall make use of Jacobi polynomials to illustrate the procedure.

Example of Approximation of a Driving Point Admittance Function by Means of Modified Jacobi Polynomials

Suppose $Y(s) = 1/\cosh s$ is a driving point admittance function. Using

O'Shea's synthesis technique, $F_1(s) = 1/[(\sqrt{s}\ \sinh\sqrt{s})(\cosh s)]$ where b=1. Then $F_1(s_1) = 1/[\cosh^{-1}s_1 \cdot \sinh(\cosh^{-1}s_1) \cdot \cosh(\cosh^{-1}s_1)^2]$ and,

$$F_2(s_2) = 1/\left[\cosh^{-1}(s_2+1)\sinh[\cosh^{-1}(s_2+1)]\cosh[\cosh^{-1}(s_2+1)]^2\right] \tag{29}$$

The approximated $F_2(s_2)$ with n=5 and α=.3 is calculated to be

$$\tilde{F}_2(s_2) = \frac{77.70}{s_2+.3} - \frac{697.35}{s_2+.6} + \frac{969.8}{s_2+.9} - \frac{1086.5}{s_2+1.2} + \frac{646.5}{s_2+1.5} - \frac{139}{s_2+1.8} \tag{30}$$

Then

$$\tilde{F}_1(s_1) = \frac{77.70}{s_1-.7} - \frac{697.35}{s_1-.4} + \frac{969.8}{s_1-.1} - \frac{1086.5}{s_1+.2} + \frac{646.5}{s_1+.5} - \frac{139}{s_1+.8} \tag{31}$$

The realization is shown in Figure 4.

B. Transfer Function Synthesis

1) Passive Network Synthesis - It is well-known[16] that the common-terminal RC networks cannot realize positive real transmission zeros. Since in general $\tilde{F}_2(s_2)$ have positive real zeros, the passive realization can be accomplished only by using balanced structure such as the lattice structure or in the form shown in Figure 5. Methods such as Guillemin[18] and Kuh[17] cannot be applied to this synthesis procedure. The following example illustrates the realization procedure using balance structure.

Example of Approximation of a Transfer Function by Means of Modified Legendre Polynomials*

For this example we will approximate the following Tchebyscheff low-pass filter

$$Z_{21}(s) = \frac{1}{s^3+1.2593s^2+1.5349s+0.7157} \tag{32}$$

After making necessary transformations we get

$$F_2(s_2) = \frac{\sqrt{s_2}\ \tanh^{-1}\sqrt{s_2}}{[\tanh^{-1}\sqrt{s_2}\,]^6+1.2593[\tanh^{-1}\sqrt{s_2}\,]^4+1.5349[\tanh^{-1}\sqrt{s_2}\,]^2+.07157} \tag{33}$$

The series expansion of the approximated $F_2(s_2)$ with n=6 and α=.055, using Legendre polynomials, will have the following coefficients

$C_0 = .06943 \quad C_1 = .2431 \quad C_2 = .05251 \quad C_3 = -.5737$

$C_4 = -1.181 \quad C_5 = -.3063 \quad C_6 = 3.6374$

The expansion of $\tilde{F}_2(s_2)$ can be expressed as

$$\tilde{F}_2(s_2) = \frac{.7273}{s_2} - \frac{60.903}{s_2+.11} + \frac{800.418}{s_2+.22} - \frac{2022.4}{s_2+.33} + \frac{8614.85}{s_2+.44} - \frac{7512.5}{s_2+.55} + \frac{2395}{s_2+.66} \tag{34}$$

The lattice realization is shown in Figure 6. The magnitude and phase of $Z_{12}(s)$ and $Z_{21}(s)$ is plotted for values of ω from 10^{-1} to 10 in Fig. 7. The region of best approximation can be shifted by suitable choice of b which is chosen as unity in this paper for the sake of simplicity.

2) Active $\overline{\text{URC}}$ Synthesis - This method is illustrated in the following example.

Example of Approximation of a Transfer Function by Means of Modified Laguerre Polynomials* Using Active Network Synthesis.

Assume given transfer function is irrational and has the following

---

*Other orthogonal polynomials can be used as well.

$$Z_{21}(s) = \frac{7.84x10^{-8}\sqrt{s}\ \tanh\sqrt{s}}{(\tanh^4\sqrt{s}+.16)^{\frac{1}{2}}} \tag{35}$$

after the transformation $F_2(s_2)$ can be obtained as

$$F_2(s_2) = 7.84x10^{-8}/(s_2^2+.16)^{\frac{1}{2}} \tag{36}$$

The truncated expansion of this function is given in Reference 11 as

$$F_2(s_2) = \frac{1.704}{s_2+.4} + \frac{3.154s_2}{(s_2+.4)^2} + \frac{.205s_2^{\,2}}{(s_2+.4)^3} - \frac{2.075s_2^{\,3}}{(s_2+.4)^4} + \frac{.38s_2^{\,4}}{(s_2+.4)^5}$$

$$+ \frac{.53s_2^{\,5}}{(s_2+.4)^6} - \frac{.754s_2^{\,6}}{(s_2+.4)^7} + \frac{.474s_2^{\,7}}{(s_2+.4)^8} - \frac{.193s_2^{\,8}}{(s_2+.4)^9} - \frac{.04s_2^{\,9}}{(s_2+.4)^{10}} \tag{37}$$

The realization of $F_2(s_2)$ is given in Figure 8. In this figure, amplifiers A and B have infinite input and output impedances. Buffer amplifiers B, have transconductance of unity. Therefore, the transfer impedance at the input of n^{th} buffer amplifier is $s^n/(s+.4)^n$. The coefficients of expansion of $F_2(s_2)$ are provided by the amplifiers A which have transconductances equal to expansion coefficients of $F_2(s_2)$.

VI. CONCLUSION

The active and passive approximation synthesis of driving point and transfer functions using uniform RC and RCG lines has been investigated. It is found that using the approximation method described in this paper, the restriction on the realizabilaity of network function by the exact synthesis techniques can be completely removed. Moreover, the synthesis procedure is considerably simplified compared to those exact synthesis. As it is seen the approximation is not unique. Therefore one can take advantage of this factor. Also the choice of α and b can be made so that the best approximation may be obtained in a desired band of frequencies. The error calculation and the determination of best approximation technique for a given network function and error specification will be published in a second paper.

REFERENCES

1. W. Happ and P. Castro, "Distributed Parameter Circuits and Microsystems Electronics," N.E.C. Proceedings, Chicago, Illinois, vol. 16 (1960), pp. 448-460.

2. M.J. Hellstrom, "Equivalent Distributed RC Networks or Transmission Lines," IRE Transactions on Circuit Theory, vol. CT-9, Sept. 1962, pp. 247-251.

3. K.W. Heizer, "Distributed R-C Networks with Rational Transfer Functions," IRE Transactions on Circuit Theory, vol. CT-9, Dec. 1962, pp. 247-251.

4. R.W. Wyndrum, Jr., "The Exact Synthesis of Distributed RC Networks," Technical Report 400-76, Department of Electrical Engineering, New York University, May 1963.

5. J.J. Kelly and M.S. Ghausi, "On the Dominant Poles of Distributed RC Networks," Technical Report 400-92, Department of Electrical Engineering, New York University, March 1964.

6. R.P. O'Shea, "Synthesis Using Distributed RC Networks," IEEE Transactions on Circuit Theory, vol. CT-12, Dec. 1965, pp. 546-554.

7. J.J. Stein, J.H. Mulligan, Jr., and S.S. Shamis, "Realization of Transfer Functions Using Uniform RC Distributed Networks with Common Ground Connections," Technical Report 400-140, Department of Electrical Engineering, New York University, June 1966.

8. J.O. Scanlan and J.D. Rhodes, "Realizability and Synthesis of a Restricted Class of Distributed RC Networks," IEEE Transactions on Circuit Theory, vol. CT-12, Dec. 1965, pp. 577-585.

9. E.N. Protonotarios and C. Wing, "Theory of Nonuniform RC Lines," IEEE Transactions on Circuit Theory, vol. 14, March 1967, pp. 1-20.

10. Y.W. Lee, "Synthesis of Electrical Networks by Means of the Fourier Transforms of Laguerre Functions," Jour. Math. Phys., vol. 11 (1932), pp. 83-113.

11. A. Papoulis, "A New Method of Inversion of Laplace Transform," Quart. Appl. Math., vol. XIV (1956), pp. 405-414.

12. H.L. Armstrong, "On the Representation of Transients by Series of Orthogonal Functions," IRE Transactions on Circuit Theory, vol. CT-6, Dec. 1959, pp. 351-354.

13. Y.W. Lee, "Statistical Theory of Communications," John Wiley, 1960.

14. N. Balabanian and S. Seshu, "Linear Network Analysis," John Wiley, 1959.

15. R. Courant and D. Hilbert, "Methods of Mathematical Physics," vol. 1, Interscience Publishers, 1953.

16. S. Seshu and M.B. Reed, "Linear Graphs and Electrical Networks," Addison-Wesley, 1961.

17. E.S. Kuh, "Synthesis of RC Grounded Two Ports," IRE Transactions on Circuit Theory, vol. CT-5 (1958), pp. 55-61.

18. E.A. Guillemin, "Synthesis of RC Networks," Jour. Math. and Phys., vol. 28 (1949), pp. 22-42.

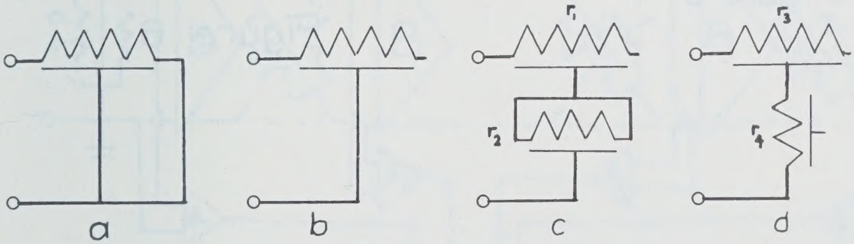


Figure 1

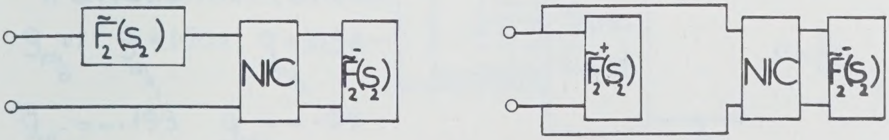


Figure 2

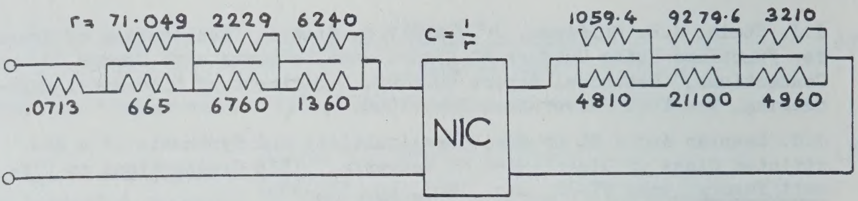

r = 71·049
2229
6240
·0713
665
6760
1360
c = 1/r
NIC
1059·4
9279·6
3210
4810
21100
4960

Figure 3

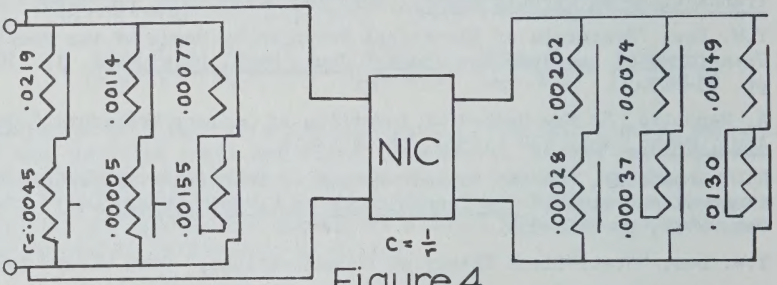

·0219
·00114
·00077
r = ·0045
·00005
·00155
NIC
c = 1/r
·00202
·00074
·00149
·00028
·00037
·0130

Figure 4

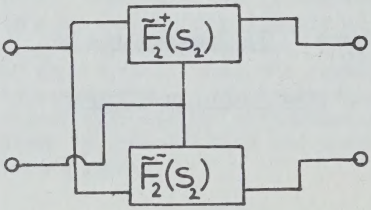

$\tilde{F}_2^+(s_2)$
$\tilde{F}_2^-(s_2)$

Figure 5

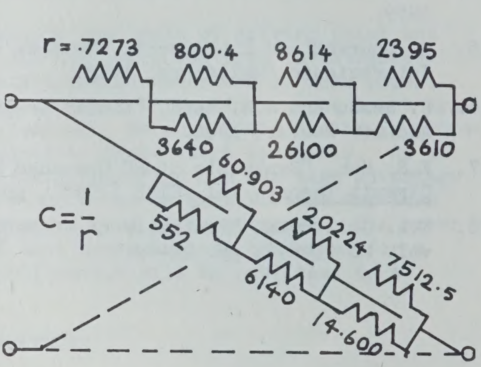

r = ·7273
800·4
8614
2395
3640
26100
3610
60·903
552
20224
7512·5
6140
14·600
C = 1/r

Figure 6

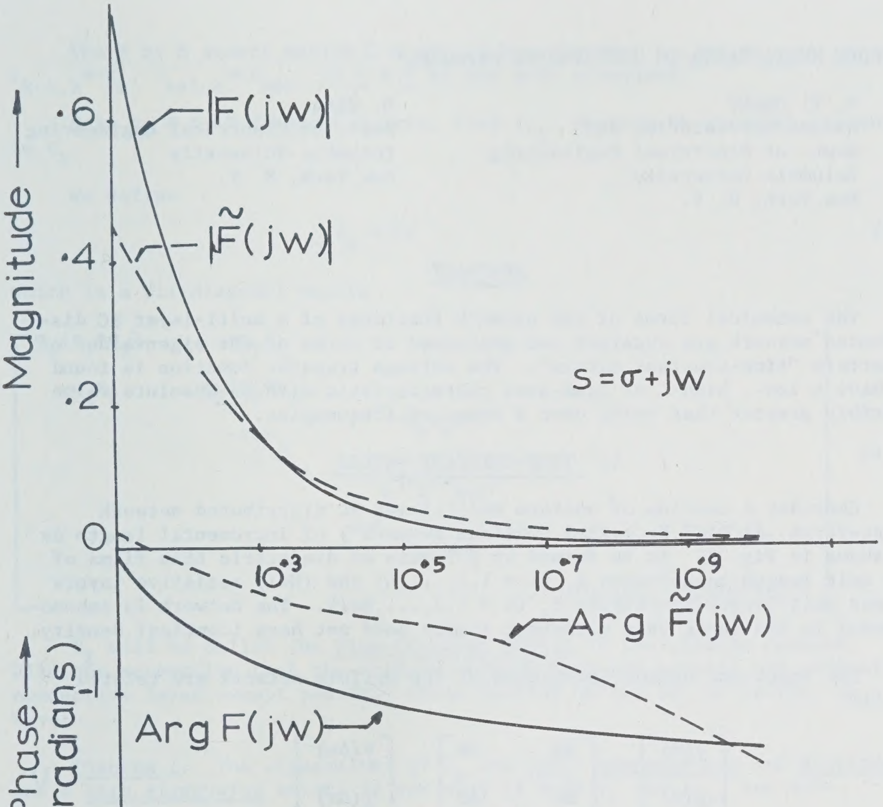


Figure 7

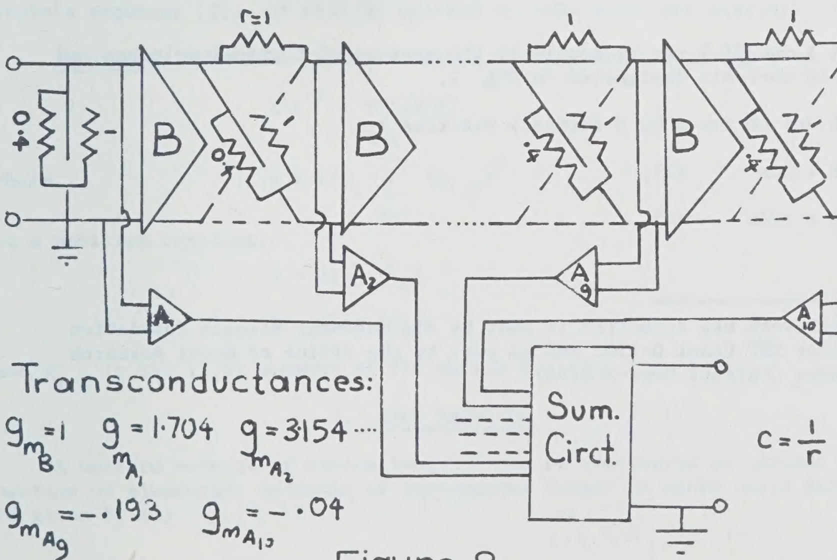

r=1
1
1
B
B
B
A_1
A_2
A_9
A_{10}
Sum. Circt.
$C = \frac{1}{r}$
Transconductances:
$g_{m_B} = 1$ $g_{m_{A_1}} = 1.704$ $g_{m_{A_2}} = 3.154$ ....
$g_{m_{A_9}} = -.193$ $g_{m_{A_{10}}} = -.04$

Figure 8

UNIFORM MULTI-LAYER RC DISTRIBUTED NETWORKS*

F. Y. CHANG
Hudson Laboratories and
Dept. of Electrical Engineering
Columbia University
New York, N. Y.

O. Wing
Dept. of Electrical Engineering
Columbia University
New York, N. Y.

ABSTRACT

The canonical forms of the network functions of a multi-layer RC distributed network are obtained and expressed in terms of the eigenvalues of a certain "time-constant matrix". The voltage transfer function is found to have a low-, high-, or band-pass characteristic with an absolute value possibly greater than unity over a range of frequencies.

1. TIME-CONSTANT MATRIX

Consider a section of uniform multi-layer RC distributed network (henceforth, it will be called "Uniform Network") of incremental length Δx as shown in Fig. 1. It is formed by N layers of dielectric thin films of per unit length capacitance c_i $(i = 1,2,\ldots,N)$ and $(N+1)$ resistive layers of per unit length resistance r_i $(i = 1,2,\ldots,N+1)$. The network is inhomogeneous in the sense that different layers need not have identical density.

The input and output quantities of the uniform network are related by [1]:

$$\begin{bmatrix} \underset{\sim}{V}(0) \\ \underset{\sim}{I}(0) \end{bmatrix} \simeq \begin{bmatrix} \Delta A & \Delta B \\ \Delta C & \Delta D \end{bmatrix} \cdot \begin{bmatrix} \underset{\sim}{V}(\Delta x) \\ \underset{\sim}{I}(\Delta x) \end{bmatrix} \qquad (1)$$

where:

a) $\underset{\sim}{V}(.)$ and $\underset{\sim}{I}(.)$ are respectively the vectors of terminal voltages and loop currents designated in Fig. 1,

b) ΔA and ΔD are N by N identity matrices I_N,

c) $\Delta B = \Omega \Delta x$ (2),

d) $\Delta C = sF\Delta x$ (3),

---

* This work was supported in part by the National Science Foundation under NSF Grant G-1505 and in part by the Office of Naval Research under Contract Nonr-266(66).

The N by N square matrix $\Omega = [\gamma_{i,j}]$ has elements $\gamma_{k,k}=r_k+r_{k+1}$ $\gamma_{k-1,k}=-r_k$, $\gamma_{k+1,k}=-r_{k+1}$ $(1 \leq k \leq N)$ and zero elsewhere.

F is an N by N diagonal matrix, diag (c_k) whose k-th diagonal element is c_k.

We define

$$T_N = \Omega F \tag{4}$$

which is a tri-diagonal matrix

$$\begin{bmatrix} c_1(r_1+r_2) & -c_2 r_2 & & & & & \\ -c_1 r_2 & c_2(r_2+r_3) & \cdot & & & & \\ & -c_2 r_3 & \cdot & -c_k r_k & & & \\ & & \cdot & c_k(r_k+r_{k+1}) & \cdot & & \\ & & & -c_k r_{k+1} & \cdot & \cdot & -c_N r_N \\ & & & & \cdot & -c_{N-1} r_N & c_N(r_N+r_{N+1}) \end{bmatrix} \tag{4'}$$

T_N will be called the Time-Constant Matrix of the uniform network. With the assumption that the uniform network does not contain any perfectly conductive layer except possibly those located on the top or bottom, we have:

Theorem 1: The eigenvalues of T_N are real, non-negative and distinct and a zero eigenvalue exists if and only if both r_1 and r_{N+1} are zero.

The proof makes use of Levy-Hadamard-Gershgorin theorem [2] and Sturm's sequence [3]. It will be omitted in this paper for brevity.

Theorem 2: The Time-Constant Matrix T_N admits the diagonal form

$$X \Lambda X^{-1} = X \Lambda (pX'F) \tag{5}$$

where

$$p = 1/c_i \sum_{k=1}^{N} (x_{i,k})^2 \tag{6}$$

is a positive constant,

$$\Lambda = \text{diag } (\lambda_k),$$

$$F = \text{diag } (c_k),$$

and $x_{i,k}$ is the (i,k) element of the matrix X.

2. ABCD MATRICES

A uniform network of finite length ℓ can be considered as cascade connection of elementary networks of incremental length Δx whose chain matrix is given by (1).

The total number of such elementary networks is $\ell/\Delta x$. Allowing $\Delta x \to 0$, the transfer matrix of the overall network can be shown to be

$$e^{\begin{bmatrix} 0 & \Omega \\ sF & 0 \end{bmatrix} \ell} \tag{7}$$

By expanding (7) in power series form and noting that the matrix $\Omega F(=T_N)$ and its transpose, $F\Omega$ are diagonable as is stated in Theorem 2, we obtain the matrices A,B,C and D as follows:

$$A=X \text{ diag } (\cosh \sqrt{\lambda_k s}\ell\ ) X^{-1} \tag{8}$$

$$B=X \text{ diag } (\sqrt{\lambda_k/s}\ \sinh \sqrt{\lambda_k s}\ell\ ) X^{-1}F^{-1} \tag{9}$$

$$C=(X')^{-1} \text{ diag } (\sqrt{s/\lambda_k}\ \sinh \sqrt{\lambda_k s}\ell\ ) X'F \tag{10}$$

$$D=(X')^{-1} \text{ diag } (\cosh \sqrt{\lambda_k s}\ell\ ) X' \tag{11}$$

where X is the set of eigenvectors associated with the eigenvalues λ_k of the time-constant matrix T_N.

3. IMMITTANCE FUNCTIONS

A uniform network with N dielectric layers is treated as a 2N-port network with the terminal voltages and loop currents of each port designated as shown in Fig. 2.

By using the same notation of voltage and current vectors introduced for the uniform network of incremental length Δx, we can express the voltages and currents of the first N ports in terms of those of the last N ports:

$$\begin{bmatrix} \underset{\sim}{V}(0) \\ \underset{\sim}{I}(0) \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \cdot \begin{bmatrix} \underset{\sim}{V}(\ell) \\ \underset{\sim}{I}(\ell) \end{bmatrix} \tag{12}$$

where ℓ is the length of the uniform network and the matrices A,B,C and D are given by equations (8) through (11).

The open-circuit impedance functions are defined for the case when all but one of the first N ports are open-circuited. The uniform network is said to be z-terminated when the last N ports (i.e. (N+K)-th port, $1 \leq k \leq N$) are connected with impedances $(c/c_k)z(s)$ $(1 \leq k \leq N)$ where c has the dimension of capacitance and z(s) is an arbitrary impedance function. The open-circuit impedance functions for this z-terminated uniform network are denoted by

$$z_{i,j}^{(z)}(s) = V_i(0)/I_j(0)$$

and

$$z_{N+i,j}^{(z)}(s) = V_i(\ell)/I_j(0) \qquad (1 \leq i,j \leq N)$$

By substituting the terminal conditions of the z-terminated network:

$$\underset{\sim}{I}(0) = \text{col}\ (0,\ldots,0,I_j(0),0,\ldots,0)$$

$$\underset{\sim}{V}(\ell) = c\ z(s)F^{-1}I(\ell)$$

into the matrix equation (12) and making use of equations (5) and (6) to simplify the matrix multiplications and inversions, one can prove the following theorem:

Theorem 3: The open-circuit impedance functions of a z-terminated uniform network admit the following canonical forms:

$$Z^{(z)}_{i,j}(s) = \left(\frac{c}{c_j}\right)\sum_{k=1}^{N} h_{i,j,k}\left[\frac{z(s) + (\sqrt{\lambda_k}\,s\ell\ \tanh\sqrt{\lambda_k}\,s\ell)/sc\ell}{1 + (sc\ell\ z(s)\ \tanh\sqrt{\lambda_k}\,s\ell)/\sqrt{\lambda_k}\,s\ell}\right] \tag{13}$$

$$Z^{(z)}_{N+i,j}(s) = \left(\frac{c}{c_j}\right)\sum_{k=1}^{N} h_{i,j,k}\left[\frac{z(s)/\cosh\sqrt{\lambda_k}\,s\ell}{1 + (sc\ell\ z(s)\ \tanh\sqrt{\lambda_k}\,s\ell)/\sqrt{\lambda_k}\,s\ell}\right] \tag{14}$$

where the coefficients $h_{i,j,k}$ $(1 \leq i,j,k \leq N)$ are real and satisfy the following constraints:

$$\text{(a).}\quad \sum_{k=1}^{N} h_{i,j,k} \begin{array}{ll} = 0 & \text{if}\quad i \neq j \\ = 1 & \qquad\ \ i=j \end{array} \tag{15}$$

$$\text{(b).}\quad \begin{array}{l} h_{i,j,k} = pc_j\,x_{i,k}\,x_{j,k} \\ h_{j,j,k} = pc_j\left(x_{j,k}\right)^2 \geq 0 \end{array} \tag{16}$$

$$\text{(c).}\quad \sum_{k=1}^{N} \left| h_{i,j,k} \right| < \sqrt{c_j/c_i} \qquad (i \neq j) \tag{17}$$

$$\text{(d).}\quad \sum_{k=1}^{N} \lambda_k h_{i,j,k} \begin{array}{lll} < 0 & & |i-j| = 1 \\ = 0 & \text{if} & |i-j| > 1 \\ > 0 & & i=j \end{array} \tag{18}$$

where p and $x_{i,k}$ are defined in Theorem 2 and use has been made of Schwarz inequality [4] to derive (c).

It can be shown that the typical term [5]:

$$\left(\frac{c}{c_j}\right) h_{j,j,k} \left[\frac{z(s) + (\sqrt{\lambda_k s}\ell \tanh \sqrt{\lambda_k s}\ell)/sc\ell}{1 + (sc\ell\, z(s) \tanh \sqrt{\lambda_k s}\ell)/\sqrt{\lambda_k s}\ell} \right]$$

of the canonical form (13) for the case, i=j, is the input impedance function of a uniform RC line terminated in an impedance $(c/c_j)\, h_{j,j,k}\, z(s)$. The total resistance and total capacitance of this single-layer RC line (i.e. only one dielectric layer) are respectively given by

$$\begin{aligned} R'_{j,k} &= r'_{j,k}\, \ell_k = h_{j,j,k}(\lambda_k \ell / c_j) \\ C'_{j,k} &= c'_{j,k}\, \ell_k = c_j \ell / h_{j,j,k} \end{aligned} \qquad (1 \le k \le N) \tag{19}$$

Hence, as far as the open-circuit input impedance function, $Z^{(z)}_{j,j}(s)$, is concerned, the uniform network with N dielectric layers is equivalent to the series connection of N uniform RC lines. This equivalence between the coupled and decoupled RC lines is illustrated in Fig. 3.

Note that if both the top and bottom resistive layers of the uniform network become perfectly conductive, then by Theorem 1, the smallest eigenvalue of T_N is zero. Hence, in view of equation (19), one of the uniform single-layer RC lines of the equivalent network degenerates to a lumped capacitor.

By using the superscripts "o" and "s" to designate respectively the open and short circuit z(s), we have the following corollary:

Corollary: The open-circuit impedance functions for the case when the last N ports of the uniform network are either all open circuit or all short circuit admit the following canonical forms:

$$Z^{(o)}_{i,j}(s) = (1/sc_j\ell) \sum_{k=1}^{N} h_{i,j,k} \sqrt{\lambda_k s}\ell / \tanh \sqrt{\lambda_k s}\ell \tag{20}$$

$$Z^{(o)}_{N+i,j}(s) = (1/sc_j\ell) \sum_{k=1}^{N} h_{i,j,k} \sqrt{\lambda_k s}\ell / \sinh \sqrt{\lambda_k s}\ell \tag{21}$$

$$Z^{(s)}_{i,j}(s) = (1/sc_j\ell) \sum_{k=1}^{N} h_{i,j,k} \sqrt{\lambda_k s}\ell \tanh \sqrt{\lambda_k s}\ell \tag{22}$$

Note that for i=j, equations (20) and (22) indicate that the input impedance across the j-th dielectric layer of the uniform network is proportional to the impedance of a lumped capacitor whose capacitance is identical to the total capacitance of the j-th dielectric layer.

Results similar to those given by (13), (14) and (20)-(22) can be obtained on an admittance basis. The input admittance has a canonical expression which leads to an equivalent network consisting of N uniform single-layer RC distributed networks connected in parallel.

4. COMMON-LAYER AND INTER-LAYER VOLTAGE TRANSFER FUNCTIONS

With reference to Fig. 2, it is clear that the voltage transfer function of a uniform network can be defined for various terminal conditions. In this paper, we will only consider the case when all of the ports at the far end are open circuit and the first N ports (except the j-th port which is connected with a voltage source) are either all short circuited or open circuited and will be designated by the superscripts "s" and "o" respectively. By imposing appropriate terminal conditions into the matrix equation (12) and having gone through some lengthy matrix manipulations, we have:

Theorem 4: The voltage transfer functions as defined above admit the following canonical forms:

$$H^{(s)}_{N+i,j}(s) = V_i(\ell)/V_j(0)\Big]_{\underset{\sim}{I}(\ell) = \underset{\sim}{0},\ \underset{\sim}{V}(0) = \text{col}\,(0,.,V_j(0),0,.,0)}$$

$$= \sum_{k=1}^{N} h_{i,j,k}\ \text{sech}\sqrt{\lambda_k s}\,\ell \qquad (23)$$

$$H^{(o)}_{N+i,j}(s) = V_i(\ell)/V_j(0)\Big]_{\underset{\sim}{I}(\ell) = \underset{\sim}{0},\ \underset{\sim}{I}(0) = \text{col}\,(0,.,\ I_j(0),0,.,0)}$$

$$= \frac{\sum_{k=1}^{N} h_{i,j,k}\sqrt{\lambda_k}\,\ell\ \text{csch}\sqrt{\lambda_k s}\,\ell}{\sum_{k=1}^{N} h_{j,j,k}\sqrt{\lambda_k}\,\ell\ \text{coth}\sqrt{\lambda_k s}\,\ell} \qquad (24)$$

where the coefficients $h_{i,j,k}$ and λ_k are defined in Theorem 3.

For the case, $i=j$, the voltage transfer function becomes the ratio of voltages across the same dielectric layer of the uniform network and hence will be called the common layer voltage transfer function. Similarly, the inter-layer voltage transfer function will be defined for the case, $i \neq j$. As will be illustrated in the examples, the common-layer voltage transfer function has low-pass characteristic while the inter-layer transfer function has the band-pass characteristic.

5. EXAMPLES

Example 1: (Low-pass characteristics)

Consider a unit-length uniform network with the resistance and capacitance per unit length respectively given by

$$\underset{\sim}{r} = \text{col}\,(r_1, r_2, \ldots, r_8, r_9) = \text{col}\,(0,1,6,15,20,15,6,1,0)$$

$$\underset{\sim}{c} = \text{col}\,(c_1, c_2, c_3, \ldots, c_7, c_8)$$

$$= \text{col}\,(7/2,\ 1/2,\ 1/6,\ 1/10,\ 1/10,\ 1/6,\ 1/2,\ 7/2)$$

The eigenvalues of the Time-Constant Matrix T_8 associated with the uniform network are the cardinal numbers:

$$0,1,2,3,4,5,6,7$$

Solving the eigenvectors and expressing T_8 in the diagonal form (5), one obtains

$$\begin{aligned} H^{(s)}_{8+1,1}(s) &= [V_1(1)/V_1(0)] \\ &= (1/128)\ [1+7 \operatorname{sech}\sqrt{s}+21 \operatorname{sech}\sqrt{2s}+35 \operatorname{sech}\sqrt{3s}+35 \operatorname{sech}\sqrt{4s} \\ &\qquad +21 \operatorname{sech}\sqrt{5s} + 7 \operatorname{sech}\sqrt{6s} + \operatorname{sech}\sqrt{7s}\ ] \end{aligned}$$

$$\begin{aligned} H^{(s)}_{8+2,1}(s) + H_{8+1,1}(s) &= \left\{[V_1(1) + V_2(1)]/V_1(0)\right\} \\ &= (1/64)\ [4+21 \operatorname{sech}\sqrt{s} + 42 \operatorname{sech}\sqrt{2s} + 35 \operatorname{sech}\sqrt{3s} \\ &\quad - 21 \operatorname{sech}\sqrt{5s} - 14 \operatorname{sech}\sqrt{6s} - 3 \operatorname{sech}\sqrt{7s}\ ] \end{aligned}$$

The low-pass characteristics of these voltage transfer functions are shown in Fig. 4. Observe that $|V_1(1)/V_1(0)|$ and $|[V_1(1) + V_2(1)]/V_1(0)|$ resemble the Butterworth and Jacobian-Elliptic characteristics respectively. For the latter case, a gain of magnitude greater than unity is obtainable for a certain range of frequencies.

Example 2: (Band-pass characteristic)

Consider the uniform network given by

$$r = \operatorname{col}(r_1,r_2,r_3) = \operatorname{col}(0,1,8/7)$$

$$c = \operatorname{col}(c_1,c_2) = \operatorname{col}(15/4,\ 7/30)$$

$$\ell = 1$$

The Time-Constant Matrix

$$\begin{bmatrix} 15/4 & -7/30 \\ -15/4 & 1/2 \end{bmatrix}$$

can be put in the diagonal form (5) where $X = \dfrac{\sqrt{15}}{60}\begin{bmatrix} \sqrt{14} & 14 \\ 15\sqrt{14} & -15 \end{bmatrix}$, $\Lambda = \begin{bmatrix} 1/4 & \\ & 4 \end{bmatrix}$ and $p = 32/105$

By substituting the parameters, p,λ_1,λ_2 and $x_{i,j}$ $(1\leq i,j\leq 2)$ into equations (16) and (23), we obtain the inter-layer voltage transfer function:

$H^{(s)}_{2+2,1}(s) = V_2(1)/V_1(0) = \operatorname{sech}\sqrt{s/4} - \operatorname{sech}\sqrt{4s}$ which has the band-pass characteristic as shown in Fig. 5.

6. CONCLUDING REMARKS

We have derived the canonical forms of various network functions of a uniform multi-layer RC distributed network. The canonical forms are expressed in terms of the eigenvalues of the Time-Constant Matrix associated with the network.

It is shown that the voltage transfer function may have a low-pass, band-pass, (or high-pass) characteristic, depending on the terminal condition. The absolute value of the transfer function may exceed unity over a range of frequencies. Finally, the canonical forms of the immittance functions lead directly to equivalent networks consisting of decoupled uniform one-layer RC lines.

REFERENCES

1. E. N. Protonotarious and O. Wing, "Delay and Rise Time of Arbitrarily Tapered RC-Transmission Lines," 1965 IEEE International Convention Record, part 7, pp. 1-6.

2. E. Bodewig, Matrix Calculus, Intersciences Publisher, New York, 1959, p. 67.

3. Hazony, Elements of Network Synthesis, Reinhold Publishing Corp., 1963, p. 46.

4. Kolmogorov and Fomin, Functional Analysis, Vol. 1: Metric and Normed Space, Graylock Press, Rochester, N.Y., 1957, p. 17.

5. R. W. Wyndrum, Jr., "The Exact Synthesis of Distributed RC Networks," Technical Report 400-76, New York University, College of Engineering, Department of Electrical Engineering, Laboratory of Electroscience Research. May 1963, p. 12.

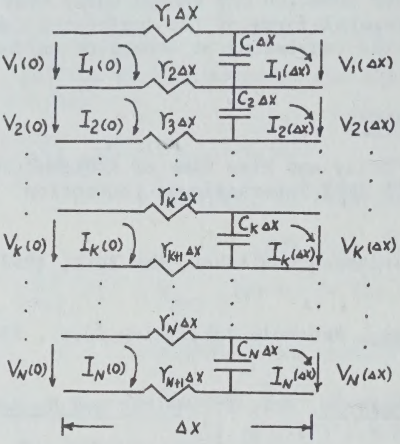

r1Δx
C1Δx
V1(0)
I1(0)
r2Δx
I1(Δx)
V1(Δx)
C2Δx
V2(0)
I2(0)
r3Δx
I2(Δx)
V2(Δx)
rKΔx
CKΔx
VK(0)
IK(0)
rK+1Δx
IK(Δx)
VK(Δx)
rNΔx
CNΔx
VN(0)
IN(0)
rN+1Δx
IN(Δx)
VN(Δx)
Δx

Fig. 1. A section of uniform multi-layer RC distributed network of incremental length Δx.

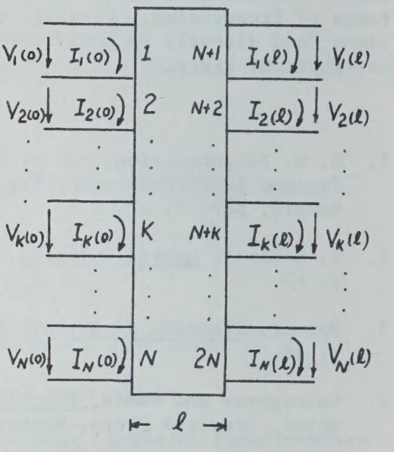

V1(0)
I1(0)
1
N+1
I1(ℓ)
V1(ℓ)
V2(0)
I2(0)
2
N+2
I2(ℓ)
V2(ℓ)
Vk(0)
Ik(0)
K
N+K
Ik(ℓ)
Vk(ℓ)
VN(0)
IN(0)
N
2N
IN(ℓ)
VN(ℓ)
ℓ

Fig. 2. A 2N-port network.

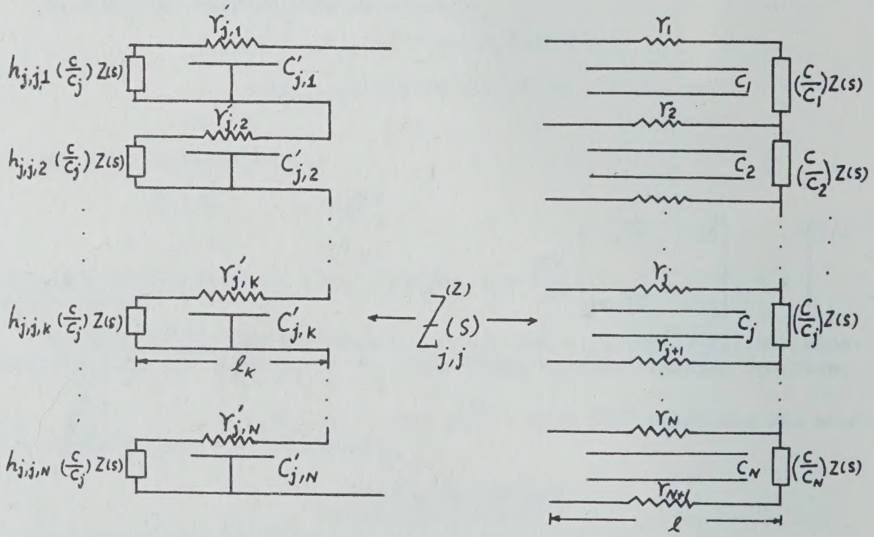

r'j,1
hj,j,1 (c/cj) Z(s)
C'j,1
r'j,2
hj,j,2 (c/cj) Z(s)
C'j,2
r'j,K
hj,j,K (c/cj) Z(s)
C'j,K
ℓK
r'j,N
hj,j,N (c/cj) Z(s)
C'j,N
Z(s) j,j
r1
C1
(C/C1)Z(s)
r2
C2
(C/C2)Z(s)
rj
Cj
(C/Cj)Z(s)
rj+1
rN
CN
(C/CN)Z(s)
rN+1
ℓ

Fig. 3. A z-terminated uniform multi-layer RC distributed network and its equivalent network corresponding to the open-circuit input impedance function.

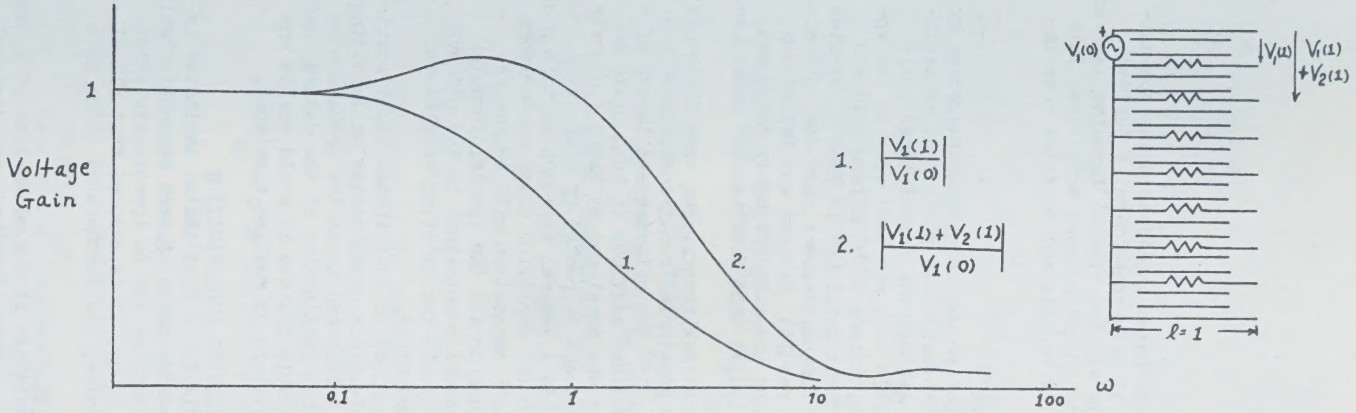


Fig. 4. The low-pass characteristics of a uniform multi-layer RC distributed network. (Example 1.)

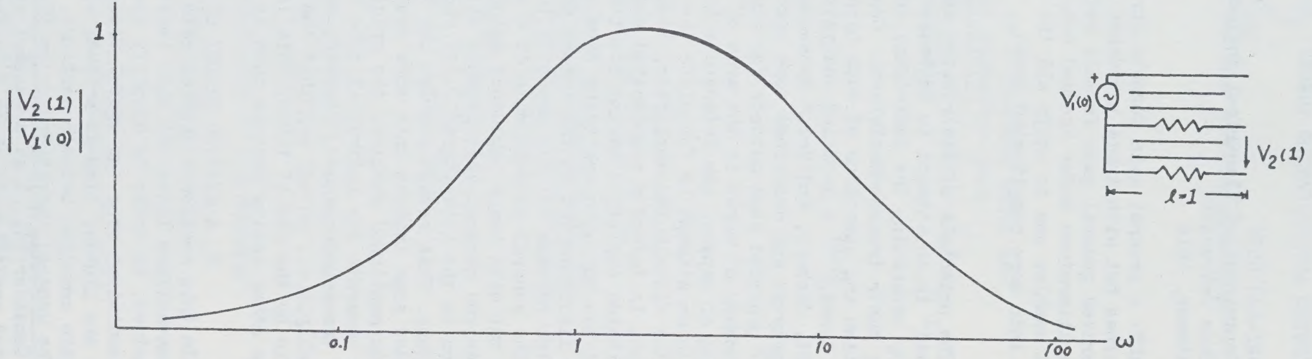


Fig. 5. The band-pass characteristic of a uniform multi-layer RC distributed network. (Example 2.)

DISTRIBUTED AMPLIFICATION THEORY

WAI-KAI CHEN
Department of Electrical Engineering
Ohio University
Athens, Ohio

ABSTRACT: A general gain formula for a distributed amplifier using non-uniform lines but with image impedance match at each connection is derived. The derived general gain formula reduces to all the special formulae derived in the literature under assumed constraints in the network structure. This result enables one to unify all the existing results and provides some insight into more complicated cases.

1. INTRODUCTION

The principle of distributed amplification was originally suggested by Percival[1] in an attempt to increase the gain-bandwidth product of an amplifier by separating the individual tube interelectrode capacitances while adding their transconductances. Considerable work has been done on the application and operation of such type of amplifiers with uniform grid and plate lines.[2-11] A detailed analysis was first published in 1948 by Ginzton, Hewlett, Jasberg, and Noe.[2] Subsequently, Horton, Jasberg, and Noe[3] extended the analysis and described some practical results. Bassett and Kelly[4] and Sarma[5] provided some methods for controlling the gain/frequency responses. For summary of works in the area of distributed amplification and extensive listing of papers, see References 6 and 12.

In an attempt to formulate the mathematical theory, one soon discovers that the distributed amplifier, seemingly so simple from transmission line concept, is indeed a complicated creature.[7-10] The mathematical theory of a distributed amplifier having identical lossless sections in both plate and grid lines but with arbitrary terminations was developed by Payne.[7] However, three discrepancies in the theory exist and are corrected by Scarr.[8]

The purpose of this paper is to provide a general approach to this problem. The general gain formula of a distributed amplifier using non-uniform lines but with image impedance match at each connection will be derived. The derived general gain formula will reduce to all the special formulae derived in the literature[2-5,7-8] under assumed constraints in the network structure. This result enables one to unify all the existing results and provides some insight into more complicated cases.

As mentioned earlier, the original purpose of distributed amplification is to separate the individual tube interelectrode capacitances while adding their transconductances. However, we shall look and treat the problem more generally, i.e., we may say that one possible realization of the desired network is by the use of tubes. This is desirable because it would enable one to use other active devices such as transistors to realize a network.

2. A UNIFIED THEORY OF DISTRIBUTED AMPLIFICATION

In this section a general gain formula for a distributed amplifier using non-uniform lines but with image impedance match at each connection will be derived. In order to simplify the mathematics and to approximate existing conditions over the useful frequency range, it is assumed that all elements are linear, time-invariant, and passive, and furthermore there is no magnetic coupling between sections.

2.1 The General Operation of the Plate-Line

Consider Fig. 1 as a general representation of the plate-line of a distributed amplifier in which n is the total number of valves per stage and $k = 1,2,\ldots,n$. Let A_k be the ABCD matrix of the kth section and also let

$$B_k = \prod_{i=1}^{k} A_i = \begin{bmatrix} b_{11}^{k} & b_{12}^{k} \\ b_{21}^{k} & b_{22}^{k} \end{bmatrix}.$$

The terminal voltages and currents of the kth section are characterized by the following equations.

$$\begin{bmatrix} X_{k-1} \\ i_{k-1} \end{bmatrix} = \begin{bmatrix} A_k & F_k \\ 0 & \delta_k \end{bmatrix} \begin{bmatrix} X_k \\ i_k \end{bmatrix},$$

where

$$X_k = \begin{bmatrix} E_{kp} \\ I_{kp} \end{bmatrix}, \quad F_k = \begin{bmatrix} a_{13k} \\ a_{23k} \end{bmatrix},$$

and δ_k is a function which relates i_{k-1} and i_k. F_k is a function which relates X_{k-1}, X_k, and i_k. It follows from mathematical induction that

$$X_o = B_n X_n + (B_{n-1} F_n + B_{n-2} F_{n-1} \delta_n + \dots + B_1 F_2 \delta_3 \dots \delta_n + F_1 \delta_2 \dots \delta_n)\, i_n .$$

Solving the equations and inserting the boundary conditions

$$Z_{rp} = E_{np}/I_{np} \quad \text{and} \quad -Z_{sp} = E_{op}/I_{op} ,$$

one has

$$E_{np} = \frac{- Z_{rp} (Z_{sp} N + M)\, i_n}{b_{11}^{n} Z_{rp} + b_{12}^{n} + b_{21}^{n} Z_{sp} Z_{rp} + b_{22}^{n} Z_{sp}}, \tag{1}$$

where

$$M = \sum_{k=1}^{n} (b_{11}^{k-1} a_{13k} + b_{12}^{k-1} a_{23k})\, \delta_{k+1} \dots \delta_n ,$$

$$N = \sum_{k=1}^{n} (b_{21}^{k-1} a_{13k} + b_{22}^{k-1} a_{23k})\, \delta_{k+1} \dots \delta_n ,$$

and B_o is defined as the unit matrix of order 2. Also let $\delta_{k+1} \dots \delta_n = 1$ for $k = n$.

Let Z_{1kp}, Z_{2kp}, and θ_{kp} represent the image impedance functions and image propagation function, respectively, of the kth section of the plate-line. If $Z_{2kp} = Z_{1(k+1)p}$ for $k = 1,2,\dots,n-1$, $Z_{rp} = Z_{2np}$, and $Z_{sp} = Z_{11p}$, then

$$B_k = \begin{bmatrix} (Z_{11p}/Z_{2kp})^{\frac{1}{2}} \cosh(\theta_{1p}+\dots+\theta_{kp}) & (Z_{11p}\, Z_{2kp})^{\frac{1}{2}} \sinh(\theta_{1p}+\dots+\theta_{kp}) \\ (1/Z_{11p} Z_{2kp})^{\frac{1}{2}} \sinh(\theta_{1p}+\dots+\theta_{kp}) & (Z_{2kp}/Z_{11p})^{\frac{1}{2}} \cosh(\theta_{1p}+\dots+\theta_{kp}) \end{bmatrix}.$$

For this case the numerator of (1) reduces to

$$M + Z_s N = \sum_{k=1}^{n} \left\{ \left(a_{13k} \left[Z_{11p}/Z_{2(k-1)p} \right]^{\frac{1}{2}} + a_{23k} \left[Z_{11p}\, Z_{2(k-1)p} \right]^{\frac{1}{2}} \right) \delta_{k+1} \dots \delta_n \cdot \exp\left[\theta_{1p} + \theta_{2p} + \dots + \theta_{(k-1)p} \right] \right\},$$

where $Z_{2op} = Z_{11p}$ and $\theta_{op} = 0$ by definition. The denominator of (1) becomes

$$b_{11}^{n} Z_r + b_{12}^{n} + b_{21}^{n} Z_{sp} Z_{rp} + b_{22}^{n} Z_{sp} = 2(Z_{11p}\, Z_{2np})^{\frac{1}{2}} \exp\left[\theta_{1p} + \theta_{2p} + \dots + \theta_{np} \right] .$$

Substituting these results in (1), one gets

$$E_{np} = -\tfrac{1}{2} Z_{2np}^{\frac{1}{2}}\, i_n \sum_{k=1}^{n} \left(\left[a_{13k} + Z_{2(k-1)p}\, a_{23k} \right] Z_{2(k-1)p}^{-\frac{1}{2}}\, \delta_{k+1} \dots \delta_n \exp\left[-\theta_{kp} - \dots - \theta_{np} \right] \right) \tag{2}$$

2.2 The Operation of the Grid-Line

Consider Fig. 2 as a general representation of the grid-line of the distributed amplifier used in Section 2.1. Again the image impedances are assumed to be matched at the junctions of the cascade sections including the terminations. V_k $(k=1,2,\dots,n)$ are used to indicate the voltages that the dependent current sources i_k of the plate-line depend, i.e., $i_k = -g_{mk} V_k$

where g_{mk} = the transconductance of the kth tube if tubes are used
= $g'_{mk}/(1 + s/\omega_{ok})$ of the kth transistor if transistors are used where g'_{mk} and ω_{ok} are parameters of the kth transistor.[6,12]

For convenience, identical valves are assumed to be used throughout this paper, i.e., $g_{mk} = g_m$ for all k. Also let Z_{1kg}, Z_{2kg}, and θ_{kg} be the image impedance functions and image propagation function of the kth section of the grid-line. Since the image impedances are matched at the junctions of the cascade sections, it follows that

$$i_n = -g_m E_{og} G_n (Z_{1ng}/Z_{11g})^{\frac{1}{2}} \exp\left[-\theta_{1g}-\theta_{2g}-\dots-\theta_{(n-1)g}\right],$$

where $G_n = V_n/E_{(n-1)g}$.

2.3 General Gain Formula

By substituting i_n just obtained in (2), one obtains the desired general gain formula for a distributed amplifier in which the image impedances are matched at the junctions of the cascade sections of both lines:

$$E_{np}/E_{og} = \tfrac{1}{2}\, g_m\, G_n\, (Z_{1ng}\, Z_{2np}/Z_{11g})^{\frac{1}{2}} \exp\left[-(\theta_{1g} + \theta_{2g} + \dots + \theta_{(n-1)g}\right]$$
$$\sum_{k=1}^{n} \left\{ (a_{13k}+a_{23k}Z_{1kp})Z_{1kp}^{-\frac{1}{2}}\ \delta_{k+1}\cdots\delta_n \exp\left[-(\theta_{kp} + \theta_{(k+1)p} + \dots + \theta_{np}\right]\right\} \tag{3}$$

where n = the number of valves, g_m = valve transcondcutance,
$G_n = V_n/E_{(n-1)g}$ (see Fig. 2), $\delta_k = V_{k-1}/V_k$ (see Fig. 2),
$\delta_{k+1}\cdots\delta_n = 1$ for $k = n$,

Z_{1kp}, Z_{2kp}, and θ_{kp} = image impedance functions and image propagation function, respectively, of the kth section of the plate-line,

Z_{1kg}, Z_{2kg}, and θ_{kg} = image impedance functions and image propagation function, respectively, of the kth section of the grid-line,

a_{13k}, and a_{23k} are defined for the kth section of the plate-line as follows: (see Fig. 1)

$$a_{13k} = -a_{12k}I_{kp}/i_k \text{ when } E_{(k-1)p} = 0 \text{ and } E_{kp} = 0\ ,$$

$$a_{23k} = -a_{21k}E_{kp}/i_k \text{ when } I_{(k-1)p} = 0 \text{ and } I_{kp} = 0\ ,$$

where $a_{12k} = E_{(k-1)p}/I_{kp}$ when $i_k = 0$ and $E_{kp} = 0$,

$a_{21k} = I_{(k-1)p}/E_{kp}$ when $i_k = 0$ and $I_{kp} = 0$.

2.4 Special Cases of the General Gain Formula

In many cases, the sections of the plate and the grid lines may be represented by Fig. 3. Then an explicit relationship between a_{13k} and a_{23k} and the network parameters can be obtained.

Suppose Z'_{1kp}, Z'_{2kp}, and θ'_{kp} represent the image impedance functions and image propagation function of the subnetwork k_{1p}, respectively, then a_{13k} and a_{23k} can be expressed in terms of these quantities after some algebraic manipulations:

$$a_{13k} = -(Z'_{1kp}Z'_{2kp})^{\frac{1}{2}} \sinh \theta'_{kp}\ , \text{ and } a_{23k} = -(Z'_{2kp}/Z'_{1kp})^{\frac{1}{2}} \cosh \theta'_{kp} \tag{4}$$

It is interesting to note that a_{13k} and a_{23k} in this case are independent of the network parameters of the subnetwork k_{2p}.

Suppose the image impedances at the junctions kk' (see Fig. 3) along the lines are also matched, and furthermore, the following conditions are satisfied, i.e.,

$Z'_{2kp} = Z'_{2p}$, $Z'_{2kg} = Z'_{2g}$; $Z_{1kp} = Z_{2kp} = Z_{op}$, $Z_{1kg} = Z_{2kg} = Z_{og}$, and

$$Z'_{1kp}\, Z'_{2kp} = Z'_{1p}\, Z'_{2p} \text{ for all } k,$$

where Z'_{1kg}, Z'_{2kg}, and θ'_{kg} represent the image impedance functions and image propagation function of the subnetwork k_{1g}, respectively. It follows that $Z_{1kp} = Z'_{1kp} = Z_{op}$, $Z_{1kg} = Z'_{1kg} = Z_{og}$; $\theta_{kp} = \theta'_{kp} + \theta''_{kp}$, $\theta_{kg} = \theta'_{kg} + \theta''_{kg}$; and $G_n = (Z'_{2g}/Z_{og})^{\frac{1}{2}} \exp\left[-\theta'_{ng}\right]$, $\delta_k = \exp\left[\theta''_{(k-1)g} + \theta'_{kg}\right]$;

where θ''_{kp}, θ'_{kg}, and θ''_{kg} represent the image propagation functions of the subnetwork k_{2p}, k_{1g}, and k_{2g}, respectively. Substituting these results in (3), the gain formula reduces to

$$E_{np}/E_{og} = -\tfrac{1}{2}g_m(Z'_{1p}Z'_{2p})^{\frac{1}{2}}(Z'_{2g}/Z_{og})^{\frac{1}{2}}\sum_{k=1}^{n}\exp\left[-\theta_{1g}-\ldots-\theta_{(k-1)g}-\theta'_{kg}-\theta''_{kp}-\theta_{(k+1)p}-\ldots-\theta_{np}\right] \tag{5}$$

where $\theta_{1g}+\ldots+\theta_{(k-1)g} = 0$ for $k = 1$, and $\theta_{(k+1)p}+\ldots+\theta_{np} = 0$ for $k = n$.

3. APPLICATION OF THE GENERAL GAIN FORMULA UNDER ASSUMED CONSTRAINTS IN THE NETWORK STRUCTURE

The general gain formula derived in Section 2 will now be applied to different types of network structures under assumed constraints. As mentioned earlier this would enable one to unify many of the existing results in the literature, and provide some insight into the more complicated cases. For convenience, tubes are assumed to be used throughout this section because the extension from tubes to transistors are quite straightforward. Since symmetrical Tee-type filter networks are of greatest practical interest, it will be treated more extensively in the following.

3.1 Symmetrical Tee-type Filter Sections

3.1.1 Gain: Suppose the plate and the grid lines are uniform but with different propagation functions and symmetrical Tee-type sections are used in both lines, then the gain $A = E_{np}/E_{og}$ can be simply obtained from (5) as follows:

$$\begin{aligned} A(s) &= -\tfrac{1}{2}g_m\left[Z_pZ'_p/(1+Z_g/4Z'_g)\right]^{\frac{1}{2}}\sum_{k=1}^{n}\exp\left[-(k-\tfrac{1}{2})\theta_g-(n-k+\tfrac{1}{2})\theta_p\right] \\ &= -\tfrac{1}{2}g_m\left[Z_pZ'_p/(1+x'^2_g)\right]^{\frac{1}{2}}\exp\left[-\tfrac{1}{2}n(\theta_g+\theta_p)\right]\sinh\tfrac{1}{2}n(\theta_g-\theta_p)/\sinh\tfrac{1}{2}(\theta_g-\theta_p) \end{aligned} \tag{6}$$

where $\frac{1}{2}Z$ and Z' = the series and the shunt branches of the symmetrical Tee-type networks, respectively. The subscripts p and g are used to indicate the plate and the grid lines, respectively; $\theta_p = \alpha_p + j\beta_p = 2\sinh^{-1}x'_p$ and $\theta_g = \alpha_g + j\beta_g = 2\sinh^{-1}x'_g$; and $x'^2_p = Z_p/4Z'_p$ and $x'^2_g = Z_g/4Z'_g$.

If constant-k LC low-pass filter sections are used for both lines and such that there is no loss in amplitude due to the lines within the passband, then the steady-state gain can be obtained from (6).

$$A(j\omega) = -\tfrac{1}{2}g_mR_{op}(1-x_g^2)^{-\frac{1}{2}}\exp\left[-j\tfrac{1}{2}n(\beta_g+\beta_p)\right]\sin\tfrac{1}{2}n(\beta_g-\beta_p)/\sin\tfrac{1}{2}(\beta_g-\beta_p) \tag{7}$$

where $Z_p = j\omega L_p$, $Z_g = j\omega L_g$; $Z'_p = 1/j\omega C_p$, $Z'_g = 1/j\omega C_g$; $\theta_p = j\beta_p = j2\sin^{-1}x_p$, $\theta_g = j\beta_g = j2\sin^{-1}x_g$; $R^2_{op} = L_p/C_p$, $R^2_{og} = L_g/C_g$; $f_{cp} = 1/\pi R_{op}C_p$, $f_{cg} = 1/\pi R_{og}C_g$; and $x'_p = j\,x_p = j\,f/f_{cp}$, $x'_g = jx_g = j\,f/f_{cg}$.

The above equation is identical to the one which has been derived by Sarma[5] except x_p is used in the radical instead of x_g. It is interesting to note that (7) may also be obtained by using the transmission line concept. (7) shows that the rising gain is due to that of the grid-line, not due to that of the plate-line as it is usually assumed in the literature when the intuitive transmission line concept is used. However, when the lines are identical, it can not tell whether the rising characteristic of gain is due to the plate-line or the grid-line, but it will show up when the lines are staggered. This result completes a lacuna in the literature. The corresponding gain characteristics of the unstaggered amplifiers are obtained by putting $\beta_g = \beta_p$ in (7).

3.1.2 The Effect of Grid Loading: Suppose constant-k LC filter sections are used in both lines and grid loading is the only source of attenuation worthy of consideration. the grid-line is no longer constant-k type filter sections. However, the gain may be obtained from (6) by letting $Z_g = j\omega L_g$ and $1/Z_g' = j\omega C_g + K\omega^2$, where $K\omega^2$ is the grid input conductance of the tube. The image propagation function θ_g of the sections of the grid-line is given by $\sinh\frac{1}{2}\theta_g = (Z_g/4Z_g')^{\frac{1}{2}} = jx_g(1-j\alpha_{og}x_g)^{\frac{1}{2}}$, where $\alpha_{og} = 2\pi f_{cg}K/C_g$ and x_g is the normalized frequency for the lossless case. Expanding the equation in terms of the real and imaginary parts of θ_g gives

$$\cosh\tfrac{1}{2}\alpha_g \sin\tfrac{1}{2}\beta_g = 2^{-\frac{1}{2}} x_g\left[1 + (\alpha_{og}^2x_g^2 + 1)^{\frac{1}{2}}\right]^{\frac{1}{2}},$$

and

$$\sinh\tfrac{1}{2}\alpha_g \cos\tfrac{1}{2}\beta_g = 2^{-\frac{1}{2}} x_g\left[(1 + \alpha_{og}^2x_g^2)^{\frac{1}{2}} - 1\right]^{\frac{1}{2}}.$$

For a general purpose pentode, say a 6AK5 tube for example, the input resistance has a value 250 ohms at a frequency of 400 MHz. This means the quantity $\alpha_{og}x_g$ has a value of the order of 0.35. Also the attenuation function $\frac{1}{2}\alpha_g$ is usually small. The following results are obtained by retaining up to first order terms in the binominal expansions of the above equations:

$$\beta_g \approx 2 \sin^{-1} x_g \text{ and } \alpha_g \approx \alpha_{og}\, x_g^2/(1 - x_g^2)^{\frac{1}{2}} \tag{8}$$

The phase-shift characteristic for this case is approximately the same as the lossless case, and the attenuation function is identical to that derived by Horton, Jasberg, and Noe.[3] Substituting these results in (6), the gain is given by

$$A(j\omega) = -\tfrac{1}{2}g_mR_{op}\left[1 - x_g^2(1 - j\alpha_{og}x_g)\right]^{-\frac{1}{2}} \exp\left[-\tfrac{1}{2}n\alpha_g - j\tfrac{1}{2}n(\beta_g+\beta_p)\right]$$
$$\sinh\tfrac{1}{2}n\left[\alpha_g+j(\beta_g-\beta_p)\right] / \sinh\tfrac{1}{2}\left[\alpha_g+j(\beta_g-\beta_p)\right] \tag{9}$$

where $K\omega^2$ is the total shunt conductance at the grid of the tube; $\alpha_{og} = 2\pi f_{cg}K/C_g$; and $\alpha_g = \alpha_{og}x_g^2/(1 - x_g^2)^{\frac{1}{2}}$. All other symbols are defined the same as in Section 3.1.1.

In the derivation of (9) it is implicitly assumed that the grid-line is terminated by its image impedances. The magnitude of (9) is identical with that obtained by Horton, Jasberg, and Noe[3] if the lines are assumed to be unstaggered, and the term $\alpha_{og}x_g$ is ignored. However, this is not always the case since $\alpha_{og}x_g$ may not be very small in comparison with unity.

3.2 Gain/Frequency and Phase/Frequency Controls

A study of (6) shows that the gain is inversely proportional to $(1+x_g'^2)^{\frac{1}{2}}$ which is the main cause for the rising characteristic near the cutoff frequency. In order to flatten out the peak, the factor $\sinh\frac{1}{2}n(\theta_g-\theta_p)/\sinh\frac{1}{2}(\theta_g-\theta_p)$ has to vanish at $x_g' = j1$. Since the staggering thus obtained is not necessarily optimum (maximum flat), it may as well consider the lossless case, and then make some proper adjustment to get the optimum value of staggering. Consequently, let $\frac{1}{2}n(\beta_g-\beta_p) = \pi$ at $x_g = 1$. Since $x_g = \sin\frac{1}{2}\beta_g$ and $x_p = \sin\frac{1}{2}\beta_p$, one obtains $\rho = \sin(\pi/2 - \pi/n)$, where $\rho = f_{cg}/f_{cp}$, and the phase-shift ϕ is given by $\phi = \frac{1}{2}n(\beta_p+\beta_g) = n(\sin^{-1}\rho x_g + \sin^{-1} x_g)$. However, the ρ obtained above is not the optimum value.

Since the plate-line usually has a higher cutoff frequency than that of the grid-line, this method does not reduce the design cutoff frequency. The difference between the phase shifts produced by the two lines increases continuously as the frequency is increased. It is obvious at this point that the steady-state characteristic of the gain can be improved. As an illustration, consider the magnitude and phase functions of (9). Using (8) in conjunction with (9), normalized curves of gain and phase-shift for various values of ρ with $\alpha_{og} = 0.2$ and $n = 5$ are plotted as shown in Fig. 4 as a function of the normalized frequency x_g. From the curves, it is obvious that the staggering of the lines will not improve gain/frequency response if α_{og} is of order 0.2 or larger.

3.3 m-derived Fileter Sections

Suppose m-derived filter sections of Tee-type networks (see Fig. 5) are used in both lines, the gain can be similarly calculated. Since the sections are no longer two-ports, (3) has to be used to obtain the gain. For this case $a_{13k} = -m_p x'_p(Z_p Z'_p)^{\frac{1}{2}}/\left[1+(1-m_p^2)x_p'^2\right]$, $a_{23k} = -1/\left[1+(1-m_p^2)x_p'^2\right]$, $\delta_k = \exp\left[\theta_g\right]$, $G_n = \left\{(1+x_g'^2)^{\frac{1}{2}}\left[(1+x_g'^2)^{\frac{1}{2}} + m_g x'_g\right]\right\}^{-1}$, $Z_{1kp} = Z_{2kp} = \left[Z_p Z'_p(1+x_p'^2)\right]^{\frac{1}{2}}$, and $Z_{1kg} = Z_{2kg} = \left[Z_g Z'_g(1+x_g'^2)\right]^{\frac{1}{2}}$ for $k = 1,2,\ldots,n$ where again the subscripts p and g are used to indicate the plate and the grid lines, respectively, and all other symbols which have not specifically defined here are the same as those defined in Section 3.1.1. Substituting these into (3), and after some mathematical manipulations, the gain is given by

$$A(s) = B_o(nZ'_a)^{-1} \exp\left[-\tfrac{1}{2}n(\theta_g+\theta_p)\right] \sinh\tfrac{1}{2}n(\theta_g-\theta_p)/\sinh\tfrac{1}{2}(\theta_g-\theta_p) \qquad (10)$$

where $B_o = -\frac{1}{2}ng_m(Z_p Z'_p)^{\frac{1}{2}}$, $\theta_p = 2\tanh^{-1}\left[m_p x'_p/(1+x_p'^2)^{\frac{1}{2}}\right]$, $\theta_g = 2\tanh^{-1}\left[m_g x'_g/(1+x_g'^2)^{\frac{1}{2}}\right]$, and $Z'_a = \left\{(1+x_g'^2)\left[1+(1-m_p^2)x_p'^2\right]\left[1+(1-m_g^2)x_g'^2\right]\right\}^{\frac{1}{2}}$.

If the prototype networks are constant-k Tee-type filter sections, the gain is still given by (10) except that B_o is replaced by A_o where $A_o = -\frac{1}{2}n g_m R_{op}$, the zero-frequency gain, and all other symbols are defined the same as in Scetion 3.1.1. The steady-state gain (replace s by $j\omega$) of the above equation reduces to the case derived by Sarma[5] when the prototype networks are constant-k LC filter sections and also the lines are assumed to be lossless, i.e.,

$$A(j\omega) = A_o Z_a \exp\left[-j\tfrac{1}{2}n(\beta_g+\beta_p)\right] \sin\tfrac{1}{2}n(\beta_g-\beta_p)/\left[n\sin\tfrac{1}{2}(\beta_g-\beta_p)\right] \qquad (11)$$

where $Z_a = \left\{(1-x_g^2)\left[1-(1-m_g^2)x_g^2\right]\left[1-(1-m_p^2)x_p^2\right]\right\}^{-\frac{1}{2}}$. It is evident that (7) can be obtained from (11) by letting $m_p = m_g = 1$. However, it cannot be done in Sarma's paper[5].

3.3.1 The Effect of Grid-loading: Again assume the m-derived LC low-pass filter sections are used in both lines and the grid loading is the only source of attenuation worthy of consideration. Following a similar argument as that given in Section 3.1.2, the image propagation function θ_g of the grid-line is obtained as follows:

$$\tanh\tfrac{1}{2}\theta_g = jm_g x_g\left\{(1-j\alpha_{og}x_g)/\left[1-x_g^2(1-j\alpha_{og}x_g)\right]\right\}^{\frac{1}{2}},$$

where α_{og} and x_g are defined the same as in Section 3.1.2. Expanding this equation in terms of the real and imaginary parts of θ_g gives

$$\cosh\tfrac{1}{2}\alpha_g \sin\tfrac{1}{2}\beta_g = 2^{-\frac{1}{2}}m_g x_g\left\{\alpha_1\alpha_o^{-1}(1+\alpha_{og}^2x_g^2)\left[(1+\alpha_1^2x_g^2)^{\frac{1}{2}} + 1\right]\right\}^{\frac{1}{2}}/(1+\alpha_1^2x_g^2)^{\frac{1}{2}},$$

$$\sinh\tfrac{1}{2}\alpha_g \cos\tfrac{1}{2}\beta_g = 2^{-\frac{1}{2}}m_g x_g\left\{\alpha_1\alpha_o^{-1}(1+\alpha_{og}^2x_g^2)\left[(1+\alpha_1^2x_g^2)^{\frac{1}{2}} - 1\right]\right\}^{\frac{1}{2}}/(1+\alpha_1^2x_g^2)^{\frac{1}{2}},$$

where $\alpha_1 = \alpha_{og}\left[1-(1-m_g^2)(1+\alpha_{og}^2x_g^2)x_g^2\right]^{-1}$. If $\alpha_{og}x_g << 1$, the following results are obtained by retaining up to first order terms in the binomial expansions of the above equations:

$$\sin\tfrac{1}{2}\beta_g \approx m_g x_g/\left[1-(1-m_g^2)x_g^2\right]^{\frac{1}{2}}, \text{ and } \alpha_g \approx \alpha_{og}m_g x_g^2\left[1-(1-m_g^2)x_g^2\right]^{-1}(1-x_g^2)^{-\frac{1}{2}} \qquad (12)$$

Again, the phase-shift characteristic for this case is approximately the same as that of the lossless case, and the approximate attenuation function is idential to that derived by Horton, Jasberg, and Noe.[3] Substituting these results in (10), the steady-state gain is given by

$$A = (A_o Z_a/n)\exp\left[-\tfrac{1}{2}n\alpha_g - j\tfrac{1}{2}n(\beta_g+\beta_p)\right]\sinh\tfrac{1}{2}n\left[\alpha_g+j(\beta_g-\beta_p)\right]/\sinh\tfrac{1}{2}\left[\alpha_g+j(\beta_g-\beta_p)\right] \qquad (13)$$

In order that (13) may be plotted for various values of m_g and m_p, it is necessary that the frequency be normalized with respect to the same quantity. Thus let $m_g x_g = m_p x_p = x$. Then

$$\sin\tfrac{1}{2}\beta_g = m_g x/\left[m_g^2-(1-m_g^2)x^2\right]^{\frac{1}{2}}, \quad \sin\tfrac{1}{2}\beta_p = m_p x/\left[m_p^2-(1-m_p^2)x^2\right]^{\frac{1}{2}},$$

$$Z_a = m_g^2 \, m_p \Big/ \left((m_g^2 - x^2)\left[m_g^2 - (1-m_g^2)x^2\right]\left[m_p^2 - (1-m_p^2)x^2\right] \right)^{\frac{1}{2}} ,$$

and $$\alpha_g = \alpha_{og} \, m_g^2 \, x^2 \Big/ \left(\left[m_g^2 - (1-m_g^2)x^2\right] (m_g^2 - x^2)^{\frac{1}{2}} \right).$$

Using these in conjunction with (13), normalized curves of gain and phase shift for various values of m_g and m_p with $n = 5$ and $\alpha_{og} = 0.052$ may be plotted as a function of x. These curves are as shown in Fig. 6.

3.4 Bridged-T Sections

The bridged-T network shown in Fig. 7 may also be used in building distributed amplifiers.[2] If the network is adjusted so that $C_2(1+a) = 4C_1$ where $a = 4L_2/L_1$, the image impedances are independent of frequency and equal to $(L_1/L_2)^{\frac{1}{2}}$. Let C_{1p} represent C_1 in Fig. 7 but used in the plate-line, and similarly represent all other elements. The following set of parameters are obtained:

$a_{13k} = -R_{op} z_p' \Big/ \left(\left[1 + (1+a_p) z_p'^2\right]^2 - z_p'^2 \right)$, $a_{23k} = -\left[1+(1+a_p)z_p'^2\right] \Big/ \left(\left[1+(1+a_p)z_p'^2\right]^2 - z_p'^2 \right)$

and $\delta_k = \exp \theta_g$ for $k = 1,2,\ldots,n$ and $a_p = 4L_{2p}/L_{1p}$. (3) reduces to the form

$$A(s) = (A_{ob}/nZ_b') \exp\left[-\tfrac{1}{2}n(\theta_g+\theta_p)\right] \sinh\tfrac{1}{2}n(\theta_g-\theta_p)/\sinh\tfrac{1}{2}(\theta_g-\theta_p) \qquad (14)$$

where $A_{ob} = -\frac{1}{2}ng_m R_{bp}$, the zero-frequency gain; $\theta_p = 2\tanh^{-1}\left(z_p' / \left[1+(1+a_p)z_p'^2\right] \right)$ $\theta_g = 2\tanh^{-1}\left(z_g' / \left[1+(1+a_g)z_g'^2\right] \right)$; $z_p' = s/\omega_{bp}$, $z_g' = s/\omega_{bg}$; $\omega_{bp}^2 = 4/L_{1p}C_{2p}$, $\omega_{bg}^2 = 4/L_{1g}C_{2g}$; $R_{bp}^2 = L_{1p}/C_{2p}$, $R_{bg}^2 = L_{1g}/C_{2g}$; $a_p = 4L_{2p}/L_{1p}$, $a_g = 4L_{2g}/L_{1g}$; and $Z_b' = \left(\left[1 + (1+a_g)z_g'^2\right]^2 - z_g'^2 \right)^{\frac{1}{2}} \left(\left[1 + (1+a_p)z_p'^2\right]^2 - z_p'^2 \right)^{\frac{1}{2}}$.

Normalized curves of steady-state gain and delay characteristics of (14) with $\theta_g = \theta_p = j\beta$ for various values of a $(= a_g = a_p)$ as a function of the normalized frequency have been plotted by Sarma[5]. A comparison of these curves shows that the flat gain and delay characteristics cannot be obtained at the same time. If an amplifier is built to give a flat delay characteristic, its gain characteristic falls continuously from its zero-frequency value as the frequency increases. The effect of staggering and grid loading tends to cause the gain characteristic to fall more rapidly.

3.5 m-derived Filter Sections with Added Resistive Elements

It can be shown[4] that the delay of an amplifier can be made almost constant over the lower 85% of the pass-band and the attenuation of gain at the same time varies in an approximately Gaussian manner with frequency if a resistor is connected in series with the capacitor in an m-derived low-pass LC filter section[4]. A typical section is as shown in Fig. 8.

Assume that the filter sections of the type shown in Fig. 8 are used in both lines, the gain of such an amplifier can easily be obtained from (3). Note that one cannot use (10) for gain merely by letting $Z' = bR_o/2 + 1/Cs$ because the current injection points are different. For this case, (see (3))

$a_{13k} + Z_{1kp}a_{23k} = -R_{op}^2/(Z_{op} - m_pL_ps/2)$, $G_n = R_{og}^2 \Big/ \left[Z_{og}(Z_{og} + m_gL_gs/2) \right]$, $Z_{op} = Z_{1kp} = Z_{2kp} = R_{op}(1 + b_pR_{op}C_ps/2 + R_{op}^2C_p^2s^2/4)^{\frac{1}{2}}$, $Z_{og} = Z_{1kg} = Z_{2kg} = R_{og}(1 + b_gR_{og}C_gs/2 + R_{og}^2C_g^2s^2/4)^{\frac{1}{2}}$, $R_{op}^2 = L_p/C_p$, $R_{og}^2 = L_g/C_g$, and $\delta_k = \exp \theta_g$ for $k = 1,2,\ldots,n$ where again the subscripts p and g are used to indicate the plate and the grid lines, respectivley. Substituting these into (3), and after some mathematical simplification, the gain is given by

$$A(s) = (A_o/nZ_c') \exp\left[-\tfrac{1}{2}n(\theta_g+\theta_p)\right] \sinh\tfrac{1}{2}n(\theta_g-\theta_p)/\sinh\tfrac{1}{2}(\theta_g-\theta_p) \qquad (15)$$

where $A_o = -\frac{1}{2}ng_mR_{op}$, the zero-frequency gain; $\theta_p = 2\tanh^{-1}\left(m_px_p'/(1+b_px_p'+x_p'^2)^{\frac{1}{2}} \right)$ $\theta_g = 2\tanh^{-1}\left(m_gx_g'/(1+b_gx_g'+x_g'^2)^{\frac{1}{2}} \right)$; $x_p' = s/\omega_{cp}$, $x_g' = s/\omega_{cg}$; $\omega_{cp}^2 = 4/L_pC_p$,

$\omega_{cg}^2 = 4/L_gC_g$; $R_{op}^2 = L_p/C_p$, $R_{og}^2 = L_g/C_g$; and
$Z_c' = (1+b_gx_g'+x_g'^2)^{\frac{1}{2}}\left[1+b_gx_g'+(1-m_g^2)x_g'^2\right]^{\frac{1}{2}}\left\{1+b_px_p'+(1-m_p^2)x_p'^2\right]^{\frac{1}{2}}$.

In case the lines are unstaggered, (15) reduces to

$$A(s) = (A_o/Z_d')\exp\left[-n\theta\right] \qquad (16)$$

where $A_o = -\frac{1}{2}ng_mR_o$, $m = m_p = m_g$, $R_o = R_{og} = R_{op}$, $b = b_g = b_p$, $x' = x_g' = x_p'$, $\theta = \theta_g = \theta_p$, and $Z_d' = (1+bx'+x'^2)^{\frac{1}{2}}\left[1 + (1-m^2)x'^2 + bx'\right]$.

3.6 Constant-Resistance Type Sections

The network of the type shown in Fig. 9 may also be used in building distributed amplifiers. If the network is adjusted so that $r^2 = L/C$, the driving-point impedance looking to the right at the points 22' with the other end terminated by Z_o, the image impedances of filter section, is given by r. This means one may terminate a resistance "r" at the points 22' of the nth section of the plate-line. Similarly, a constant resistance may also be terminated at the points of the mid-shunt arm of the first section of the grid-line. Again the subscripts p and g are used to indicate the plate-line and the grid-line, respectively.

Suppose identical filter sections of the type shown in Fig. 9 are used in both lines, the gain A" $(=E_{np}'/V_1)$ can be easily obtained from (6). (See Fig. 3 for definitions of E_{np}' and V_1.) Note that the voltage gain defined here is different from those defined in the previous sections.

$$A''(s) = (A_{or}/n)(1+\tfrac{1}{2}y_p')^{-1}\exp\left[-\tfrac{1}{2}(n-1)(\theta_g+\theta_p)\right]\sinh\tfrac{1}{2}n(\theta_g-\theta_p)/\sinh\tfrac{1}{2}(\theta_g-\theta_p) \qquad (17)$$

where $A_{or} = -\frac{1}{2}ng_mr_p$, the zero-frequency gain; $\theta_g = 2\sinh^{-1}\left|\frac{1}{2}y_g'/(1+y_g')^{\frac{1}{2}}\right|$, $\theta_p = 2\sinh^{-1}\left|\frac{1}{2}y_p'/(1+y_p')^{\frac{1}{2}}\right|$; $y_g' = s/\omega_{rg}$, $y_p' = s/\omega_{rp}$; $\omega_{rg}^2 = 1/L_gC_g$, $\omega_{rp}^2 = 1/L_pC_p$; and $r_g^2 = L_g/C_g$, $r_p^2 = L_p/C_p$.

When the lines are unstaggered, (17) becomes

$$A''(s) = A_{or}(1+y')^{1-n}(1+\tfrac{1}{2}y')^{-1} \qquad (18)$$

where $y' = y_p' = y_g'$. Obviously, there is an upper limit on the number of valves that can be used economically.

3.7 Combination of m-derived Constant-k LC Type and Constant-Resistance Type Filter Sections

A study of (17) shows that the steady-state magnitude decreases continuously as the normalized frequency y is increased. In order to increase the gain to offset the attenuation caused by the resistive elements, a distributed amplifier may be constructed by using m-derived constant-k LC filter sections in the grid-line and constant-resistance filter sections of the type shown in Fig. 9 in the plate-line. The gain A' $(=E_{np}'/E_{op})$ can be obtained from (3) and is given by

$$A'(s) = (A_{oc}/nZ_{mr}')\exp\left[-\tfrac{1}{2}(n-1)(\theta_g+\theta_p) - \tfrac{1}{2}\theta_g\right]\sinh\tfrac{1}{2}n(\theta_g-\theta_p)/\sinh\tfrac{1}{2}(\theta_g-\theta_p) \qquad (19)$$

where $A_{oc} = -\frac{1}{2}ng_mr_p$, zero-frequency gain; $\theta_g = 2\tanh^{-1}\left|m_gx_g'/(1+x_g'^2)^{\frac{1}{2}}\right|$, $\theta_p = 2\sinh^{-1}\left|\frac{1}{2}y_p'/(1+y_p')^{\frac{1}{2}}\right|$; and $Z_{mr}' = (1+x_g'^2)^{\frac{1}{2}}\left[1 + (1-m_g^2)x_g'^2\right]^{\frac{1}{2}}(1+y_p')$. m_g and m_p are defined the same as in Section 3.3, and r_p and y_p' in Section 3.6.

4. CONCLUSIONS

General formula for gain of a distributed amplifier using non-uniform lines but with image impedance match at each connection is derived. It reduces to all special formulae derived in the literature for special kinds of networks used and special conditions assumed. This result enables one to unify many of the existing results and provides some insight into the more complicated cases.

5. ACKNOWLEDGMENT

The author wishes to express his thanks to his wife, Shirley S.L. Chen, for the numerical computations whose results provided many of the curves illustrating this paper.

6. REFERENCES

1. W. S. Percival, "Thermionic valve circuits," British Patent No. 460,562; January, 1937.

2. E. L. Ginzton, W. R. Hewlett, J. H. Jasberg, and J. D. Noe, "Distributed amplification," Proc. IRE, vol. 36, pp. 956-969; August, 1948.

3. W. H. Horton, J. H. Jasberg, and J. D. Noe, "Distributed amplifiers: Practical considerations and experimental results," Proc. IRE, vol. 38, pp. 748-753; July, 1950.

4. H. G. Bassett, and L. C. Kelly, "Distributed amplifiers: Some new methods for controlling gain/frequency and transient responses of amplifiers having moderate bandwidths," Proc. IEE, vol. 101, Part III, pp. 5-14; 1954.

5. D. G. Sarma, "On distributed amplification," Proc. IEE, vol. 102B, pp. 689-697; 1955.

6. W. K. Chen, "Distributed amplifier theory and design," Tech. Rep. No. EED 10-6, Department of Electrical Engineering, Ohio University, Athens, Ohio; April, 1966.

7. D. V. Payne, "Distributed amplifier theory," Proc. IRE, vol. 41, pp. 759-762; June, 1953.

8. R. W. A. Scarr, "Discussion on distributed amplifier theory," Proc. IRE, vol. 42, pp. 596-598; March, 1954.

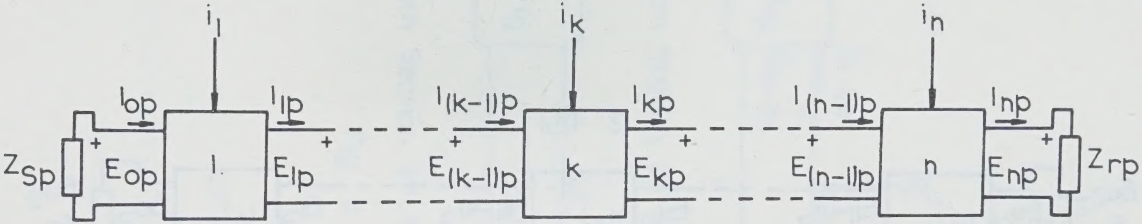
9. N. DeClaris, "Synthesis of wideband amplifiers," Proc. Symp. Active Networks and Feedback Systems, Polytech. Inst. Brooklyn, N.Y., pp. 345-367; April, 1960.

10. J. M. Pettit, and D. O. Pederson, "Distributed amplifier theory," Proc. Symp. Modern Network Synthesis, Polytech. Inst. Brooklyn, N.Y., pp. 202-214; April, 1952.

11. W. K. Chen, "Distributed amplification: A new approach," IEEE Trans. on Electron Devices, vol. ED-14, pp. 215-221; April, 1967.

12. W. K. Chen, "Distributed amplifiers: Survey of the effects of lumped-transmission-line design on performance," Proc. IEE, vol. 114, pp. 1065-1074; August, 1967.

13. W. K. Chen, "A new approach to the design of distributed amplifiers," Proc. First Annual Princeton Conference on Information Sciences and Systems, Princeton University, Princeton, N.J., pp. 145-149; March, 1967.



i1
ik
in
Iop
I1p
I(k-1)p
Ikp
I(n-1)p
Inp
ZSp
Eop
1
E1p
E(k-1)p
k
Ekp
E(n-1)p
n
Enp
Zrp
+

Fig. 1

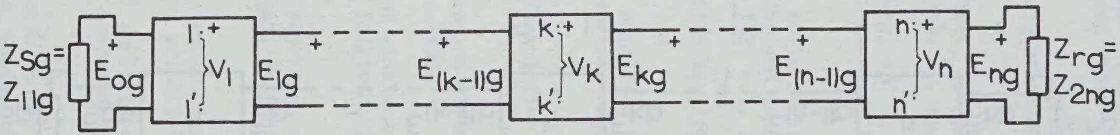
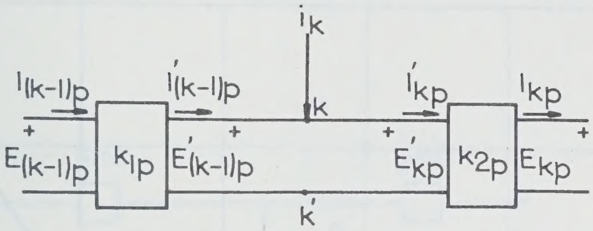
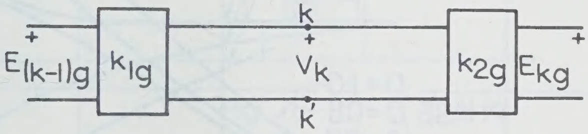

Zsg=
Z11g
Eog
1
1'
V1
E1g
E(k-1)g
k
k'
Vk
Ekg
E(n-1)g
n
n'
Vn
Eng
Zrg=
Z2ng
+

Fig. 2



i_k
$I_{(k-1)p}$
$I'_{(k-1)p}$
k
I'_{kp}
I_{kp}
+
+
+
+
$E_{(k-1)p}$
k_{1p}
$E'_{(k-1)p}$
E'_{kp}
k_{2p}
E_{kp}
k'

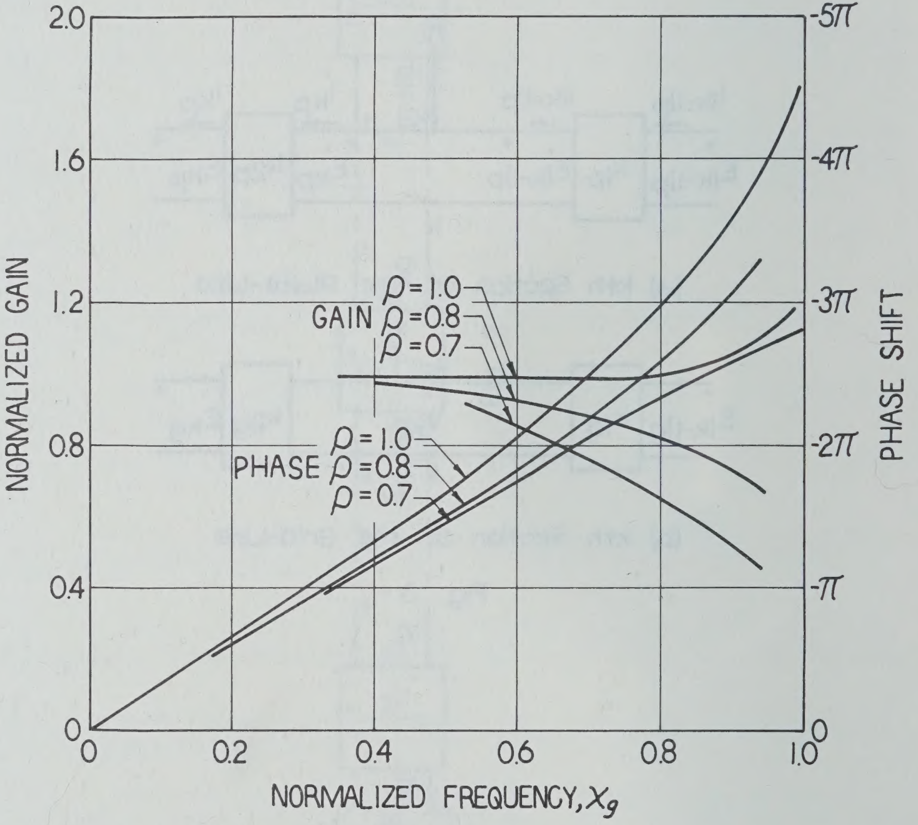
(a) kth Section of the Plate-Line



k
+
+
+
$E_{(k-1)g}$
k_{1g}
V_k
k_{2g}
E_{kg}
k'

(b) kth Section of the Grid-Line

Fig. 3



NORMALIZED GAIN
2.0
1.6
1.2
0.8
0.4
0
0
0.2
0.4
0.6
0.8
1.0
NORMALIZED FREQUENCY, X_g
-5π
-4π
-3π
-2π
-π
0
PHASE SHIFT
GAIN
ρ = 1.0
ρ = 0.8
ρ = 0.7
PHASE
ρ = 1.0
ρ = 0.8
ρ = 0.7

Fig. 4

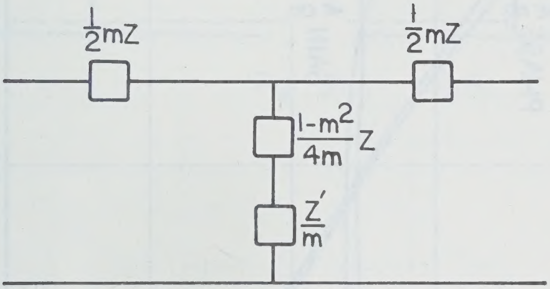

$\frac{1}{2}mZ$
$\frac{1}{2}mZ$
$\frac{1-m^2}{4m}Z$
$\frac{Z'}{m}$

Fig. 5

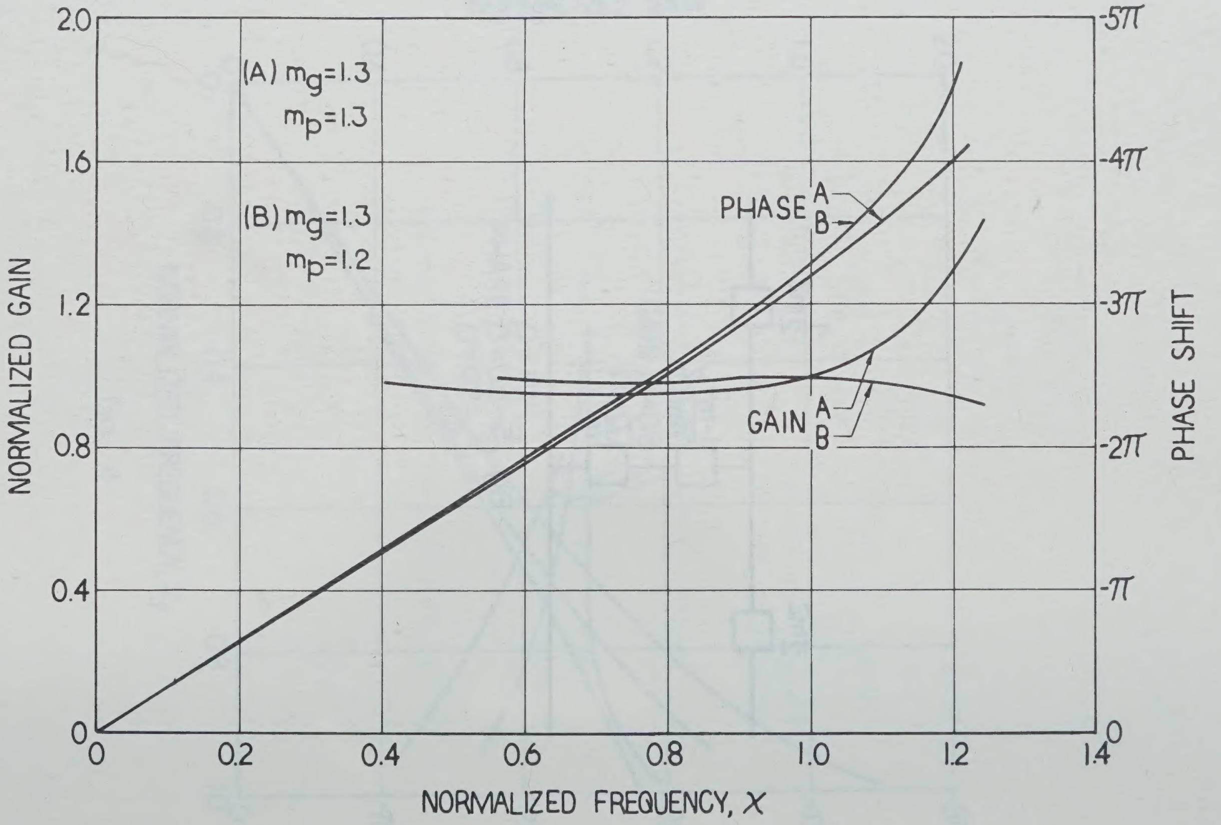

(A) $m_g=1.3$
$m_p=1.3$
(B) $m_g=1.3$
$m_p=1.2$
PHASE A
PHASE B
GAIN A
GAIN B
NORMALIZED GAIN
2.0
1.6
1.2
0.8
0.4
0
PHASE SHIFT
-5π
-4π
-3π
-2π
$-\pi$
0
0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
NORMALIZED FREQUENCY, X

Fig. 6

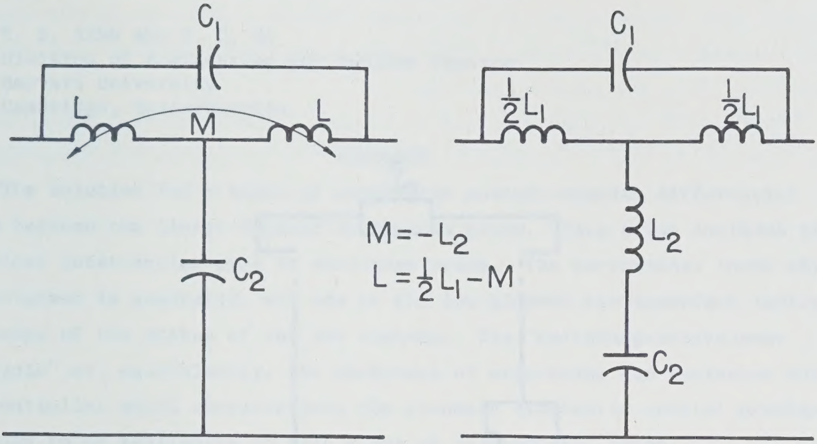

C1
L
M
L
C2
M = -L2
L = ½L1 - M
C1
½L1
½L1
L2
C2

Fig. 7

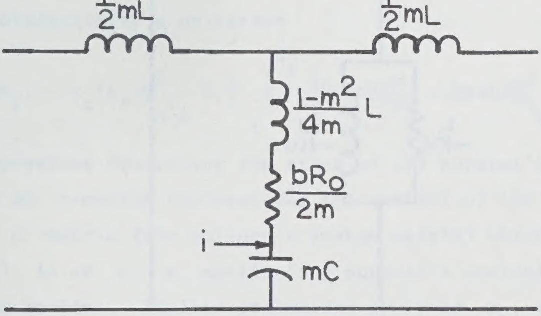
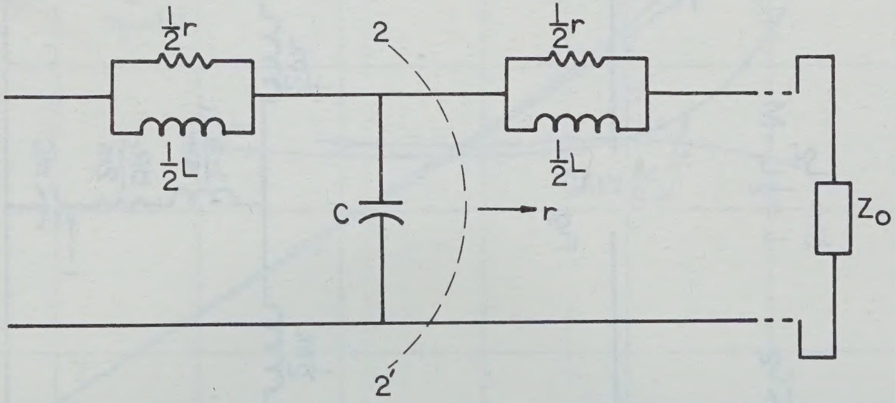

½mL
½mL
(1-m²)/4m L
bR0/2m
i
mC

Fig. 8



$\frac{1}{2}r$
$\frac{1}{2}L$
2
2'
C
r
$\frac{1}{2}r$
$\frac{1}{2}L$
Z_o

Fig. 9

ON A CLASS OF LINEAR STOCHASTIC DIFFERENTIAL GAMES

R. D. BEHN AND Y. C. HO
Division of Engineering and Applied Physics
Harvard University
Cambridge, Massachusetts

ABSTRACT

The solution for a class of stochastic pursuit-evasion differential games between two linear dynamic systems is given. This class includes the classical interception game in euclidean space. The performance index which is optimized is quadratic, and one of the two players has imperfect (noisy) knowledge of the states of the two systems. The "certainty-equivalence principle" or, equivalently, the technique of separating the estimator and the controller which characterizes the standard stochastic control problem, is shown to be applicable to this class of differential games.

INTRODUCTION AND FORMULATION OF THE PROBLEM

In 1965 Ho, Bryson, and Baron [1] used variational techniques to solve a general linear pursuit-evasion game in phase space. A pursuer and an evader each control their own linear dynamic system

$$\dot{x}_p = F_p(t)x_p + \bar{G}_p(t)u(t) \quad , \quad x_p(t_o) = x_{po} \tag{1}$$

$$\dot{x}_e = F_e(t)x_e + \bar{G}_e(t)v(t) \quad , \quad x_e(t_o) = x_{eo} \tag{2}$$

and attempted to minimaximize a criterion

$$J = \frac{a^2}{2} \|x_p(t_f) - x_e(t_f)\|^2_{A^TA} + \frac{1}{2}\int_{t_o}^{t_f} \left[\|u(t)\|^2_{R_p} - \|v(t)\|^2_{R_e}\right]dt \tag{3}$$

Here x_p is an n-vector describing the state of the pursuer's dynamic system, $u(t)$ is an m-vector representing the control of the pursuer, $F_p(t)$ is an $n \times n$ matrix (the pursuer's system matrix) which is continuous in t, and $\bar{G}_p(t)$ is an $n \times m$ matrix (the pursuer's control matrix) which is also continuous in time. Similar statements apply to x_e, $v(t)$, $F_e(t)$, and $\bar{G}_e(t)$, which describe the evader's dynamic system. $R_p(t)$ and $R_e(t)$ are respectively $m \times m$ and $m' \times m'$ dimensional, positive-definite, control weighting matrices. The matrix A is of the form $[I_k : 0]$, where I_k is the k-dimensional identity matrix, with $1 \leq k \leq n$.

The matrix A indicates which components of the two state vectors are relevant to the pursuit and evasion. The a^2 term permits a weighing of the importance of terminal miss against control effort. The game is of fixed duration, $t_f - t_o$. $u(t)$ and $v(t)$ belong to the m-dimensional and m'-dimensional euclidean spaces respectively.

The pursuer attempts to minimize the criterion (3), i.e., minimize the terminal miss distance and its own energy expended, while maximizing the energy expended by the evader. The evader attempts to maximize this same criterion. Thus the game is zero sum.

Each player is assumed to have knowledge of the other's capabilities, i.e., the F, $\bar{G}$, and R matrices are known. For the deterministic problem, each player is also assumed to have knowledge of the other's state. Since this is a zero-sum game of perfect information, optimal strategies can be determined.

These strategies are determined by first defining a reduced state vector, $y(t)$, which represents the terminal miss, $A[x_p(t_f) - x_e(t_f)]$, predicted at time t on the basis that no control will be applied during the interval (t,t_f).

$$y(t) = A[\Phi_p(t_f,t)x_p(t) - \Phi_e(t_f,t)x_e(t)] \tag{4}$$

Here Φ_p and Φ_e are the transition matrices obtained from F_p and F_e respectively. It should be noted that $y(t)$ is, in itself, more significant than $x_p(t)$ and $x_e(t)$ for $y(t)$ directly relates to the terminal miss term in the criterion.

Using (4), one can reduce (1), (2), and (3) to a simpler definition of the problem. The criterion is:

$$J = \frac{a^2}{2} \|y(t_f)\|^2 + \frac{1}{2} \int_{t_o}^{t_f} \left[\|u(t)\|^2_{R_p} - \|v(t)\|^2_{R_e}\right] dt \tag{5}$$

subject to the differential constraint

$$\dot{y}(t) = G_p(t_f,t)u(t) - G_e(t_f,t)v(t) \quad , \quad y(t_o) = y_o \tag{6}$$

where the new control matrix for the pursuer is given by

$$G_p(t_f,t) = A\Phi_p(t_f,t)\bar{G}_p(t) \tag{7}$$

and similarly for G_e. By use of its control, $u(t)$, the pursuer still attempts to minimize this new form of the criterion, while the evader, with $v(t)$, to maximize it.

The solution to the problem of (5) and (6), as obtained in [1], is given by the following feedback strategies:

$$u^o(t) = -R_p^{-1}(t)G_p^T(t_f,t)K^{-1}(t_f,t)y(t) \tag{8}$$

$$v^o(t) = -R_e^{-1}(t)G_e^T(t_f,t)K^{-1}(t_f,t)y(t) \tag{9}$$

where

$$K(t_f,t) = \frac{I_k}{a^2} + M_p(t_f,t) - M_e(t_f,t) \tag{10}$$

and

$$M_p(t_f,t) = \int_t^{t_f} G_p(t_f,\tau)R_p^{-1}(\tau)G_p^T(t_f,\tau)d\tau \tag{11}$$

A similar expression defines $M_e(t_f,t)$. It should be noted that M_p and M_e are, from control theory, the reduced controllability matrices of the pursuer and evader respectively.

It is shown in [1] that the existence of $K^{-1}(t_f,t)$ in the interval $t_o \leq t < t_f$ is equivalent to the nonexistence of a conjugate point to the game problem in this same interval. The existence of K^{-1} is guaranteed if

$$M_r(t_f,t) \overset{\Delta}{=} M_p(t_f,t) - M_e(t_f,t) > 0 \quad . \tag{12}$$

M_r can be called the "relative controllability matrix"; the fact that it is positive definite indicates that the pursuer is "more controlable" than the evader in every euclidean dimension.

A logical extension of this work is an investigation of a pursuit-evasion problem, in which the players have imperfect knowledge of the state involved. In the particular problem solved here, the pursuer is assumed to have perfect knowledge of the state involved, i.e., both $x_p(t)$ and $x_e(t)$, while the evader makes imperfect measurements of the form:

$$z_1 = x_p(t) + w_1(t) \tag{13}$$

$$z_2 = x_e(t) + w_2(t) \tag{14}$$

where w_1 and w_2 are Gaussian white noise vectors, with zero mean and covariance matrices $Q_1(t)$ and $O_2(t)$ respectively. The cross correlation between w_1 and w_2 is $C(t)$.

Since the dynamical system for the pursuit-evasion problem is given by (6), the evader's measurements must be reduced to measurements on $y(t)$,

$$z = H(t)y(t) + w(t) \tag{15}$$

where $w(t)$ is Gaussian white noise with zero mean, and covariance matrix $Q(t)$.

By premultiplying (13) by $A\Phi_p$, (14) by $A\Phi_c$, and subtracting, we can define a new measurement vector:

$$z \overset{\Delta}{=} A\Phi_p z_1 - A\Phi_e z_2 \tag{16}$$

which means z is of the form

$$z(t) = y(t) + w(t) \tag{17}$$

where

$$w(t) = A\left[\Phi_p w_1 - \Phi_e w_2\right] \tag{18}$$

has zero mean and variance

(19) $$Q = A\left[\Phi_p Q_1 \Phi_p^T + \Phi_p Q_2 \Phi_p^T - \Phi_p C \Phi_e^T - \Phi_e C^T \Phi_p^T\right] A^T \tag{19}$$

In the solution obtained below it will be assumed that the measurements are of the general form (15) rather than the more limited case of (17).

For any stochastic problem, where the objective is as explained above, the form of the criterion given in (5) is inappropriate because it assumes that $y(t_f)$ is known or can be determined exactly. Thus, this criterion must be modified by use of an expected value operator.

$$J = E\left\{\frac{a^2}{2}\|y(t_f)\|^2 + \frac{1}{2}\int_{t_o}^{t_f}\left[\|u(t)\|_{R_p}^2 - \|v(t)\|_{R_e}^2\right]dt\right\} \tag{20}$$

THE SOLUTION

The stochastic pursuit-evasion problem consists of the optimization of (20) subject to (6) with the pursuer having perfect knowledge of the state, $y(t)$, and the evader making noisy measurements on the state of the form given in (15). Optimization for a pursuit-evasion problem means that the pursuer minimizes the criterion, J, while the evader maximizes it. The solution is obtained when the optimal strategies, $u^o(t)$ and $v^o(t)$, produce a saddle point as defined by:

$$J(u^o,v) \leq J(u^o,v^o) \leq J(u,v^o) \tag{21}$$

where u^o and v^o are feedback strategies utilizing all available information.

We shall first assume the general form of the feedback relations that produce the optimal control vectors. Let the evader's feedback strategy be of the form:

$$v(t) = C_e(t)\hat{y}(t) \tag{22}$$

where $\hat{y}(t)$ is the evader's estimate of the state. The pursuer consequently should utilize a feedback strategy which depends not only on the state, $y(t)$, but also upon the error in the evader's estimate,

$$\tilde{y}(t) = y(t) - \hat{y}(t) \tag{23}$$

Thus the form of the pursuer's control is:

$$u(t) = C_p(t)y(t) + D_p(t)\tilde{y}(t) \tag{24}$$

It is shown in [2] that these strategies, with the optimal values of the feedback matrices, $C_p(t)$, $C_e(t)$, and $D_p(t)$, do provide a saddle-point in the sense of (21) for the class of all admissible strategies and thus represent the solution to the game.*

Since the pursuer is assumed only to have perfect knowledge of the state, $y(t)$, and no knowledge of the evader's estimate, $\hat{y}(t)$, the pursuer cannot in general obtain the error in the estimate, $\tilde{y}(t)$. However, substituting (22) into (6) we have:

$$G_e(t_f,t)C_e(t)\hat{y}(t) = G_p(t_f,t)u(t) - \dot{y}(t) \tag{25}$$

Since the pursuer knows $G_p(t_f,t)$, $u(t)$, and $y(t)$ perfectly, it can—by differentiating $y(t)$ with respect to time—calculate $v(t) = G_e(t_f,t)\cdot C_e(t)\cdot\hat{y}(t)$. The evader's estimate (and consequently the error in this estimate) can be calculated by the pursuer, if the inverse of $G_e(t_f,t)\cdot C_e(t)$ exists. Two necessary conditions for the existence of this inverse are:

i. The number of "interesting" state variables (i.e., the dimension of $y(t)$) must be less than or equal to the number of evader's control variables (i.e., the dimension of $v(t)$); and

ii. The inverse of $G_e(t_f,t)$ must exist for all values of $t < t_f$.

It will be shown that these two conditions** also guarantee that the inverse of $C_e(t)$ exists.

These two conditions do restrict the class of problems to which the solution obtained below is applicable though it does include the important interception problem in euclidean space; the difficulties which are encountered when these two restrictions are removed are discussed later.

Using (22) and (24) to determine the values of $u(t)$ and $v(t)$ and using (23) to eliminate $\hat{y}(t)$, the criterion (13) becomes

$$J = E\left\{\frac{a^2}{2}\|y(t)\|^2 + \frac{1}{2}\int_{t_o}^{t_f}\left[\|y(t)\|^2_{C_p^TR_pC_p} + y^T(t)C_p^TR_pD_p\tilde{y}(t) + \tilde{y}(t)D_p^TR_pC_py(t) + \|\tilde{y}(t)\|^2_{D_p^TR_pD_p} - \|y(t) - \tilde{y}(t)\|^2_{C_e^TR_eC_e}\right]dt\right\} \tag{26}$$

which is to be minimaximized subject to the dynamical constraint

$$\dot{y}(t) = \left[G_pC_p - G_eC_e\right]y(t) + \left[G_pD_p + G_eC_e\right]\tilde{y}(t) \quad , \quad y(t_o) = y_o \tag{27}$$

Since the forms of $u(t)$ and $v(t)$ are specified, the evader's estimate of the state is obtained by using a Kalman filter based on the dynamical

---

[*] The admissible class of strategies are defined by $u(t;y(\tau)\ ,\ t_o \le \tau \le t)$, and $v(t;z(\tau)\ ,\ t_o \le \tau \le t)$.

[**] These two conditions can be expressed by the single condition, rank $(G_e) \ge \dim(y)$.

equation (27)

$$\dot{\hat{y}}(t) = \left[G_pC_p - G_eC_e\right]\hat{y}(t) + PH^TQ^{-1}\left[z - H\hat{y}(t)\right] \quad , \quad \hat{y}(t_o) = \hat{y}_o \tag{28}$$

where

$$P(t) \triangleq E\left[\tilde{y}(t)\cdot\tilde{y}^T(t)\right] \equiv E\left[y(t)\cdot\tilde{y}^T(t)\right] \tag{29}$$

is the covariance of the error of the estimate.* P(t) is obtained from:

$$\dot{P}(t) = \left[G_pC_p+G_pD_p\right]P(t) + P(t)\left[G_pC_p+G_pD_p\right]^T - P(t)H^TQ^{-1}HP(t) \quad , \quad P(t_o) = P_o \tag{30}$$

and thus can be calculated and stored before the actual play of the game, once C_p, C_e, and D_p are determined.

The criterion can be further rewritten in the form

$$J = \mathrm{Tr}\left\{\frac{a^2}{2}Y(t_f) + \frac{1}{2}\int_{t_o}^{t_f}\left[Y(t)(C_p^TR_pC_p-C_e^TR_eC_e)\right.\right.$$
$$\left.\left. + P(t)(C_p^TR_pD_p+D_p^TR_pC_p+D_p^TR_pD_p+C_e^TR_eC_e)\right]dt\right\} \tag{31}$$

where

$$Y(t) \triangleq E\left[y(t)\cdot y^T(t)\right] . \tag{32}$$

Y(t) is obtained from

$$\dot{Y}(t) = G_pC_pY + YC_p^TG_p^T + G_pD_pP + PD_p^TG_p^T - G_eC_e\left[Y-P\right] - \left[Y-P\right]C_e^TG_e^T \tag{33}$$

The stochastic problem has now been transformed into a deterministic one by use of the expected value operator. For this problem, P(t) and Y(t) are the new state variables, $C_p(t)$, $D_p(t)$, and $C_e(t)$ the new control variables, (30) and (33) the dynamical system equations, and (31) is the deterministic criterion. Adjoining (33) and (30) to (31) with multiplier matrices $\Lambda_1(t)$ and $\Lambda_2(t)$ respectively, the Hamiltonian, $\mathcal{H}$, for the problem can be defined.

$$\mathcal{H} \triangleq \mathrm{Tr}\left\{\frac{1}{2}Y\left[C_p^TR_pC_p-C_e^TR_eC_e\right] + \frac{1}{2}P\left[C_p^TR_pD_p+D_p^TR_pC_p+D_p^TR_pD_p+C_e^TR_eC_e\right] + \Lambda_1\dot{Y} + \Lambda_2\dot{P}\right\} \tag{34}$$

A set of necessary conditions for a saddle point of the problem as formulated are:

---

*From (23) it can be seen that any two of the three variables $[y(t), \hat{y}(t), \tilde{y}(t)]$ completely describe the state of the system. The selection of y(t) and $\tilde{y}(t)$ leads directly to results with useful interpretations. Equation (28) may be more obvious, however, if (27) is rewritten in the form:

$$\dot{y}(t) = \left[G_pC_p + G_pD_p\right]y(t) - \left[G_eC_e + G_pD_p\right]\hat{y}(t) \quad , \quad y(t_o) = y_o$$

$$\frac{\partial\mathcal{H}}{\partial C_p} = 0 \tag{35}$$

$$\frac{\partial\mathcal{H}}{\partial D_p} = 0 \tag{36}$$

$$\frac{\partial\mathcal{H}}{\partial C_e} = 0 \tag{37}$$

The solution of the two simultaneous equations (36) and (37) is

$$C_p = -R_p^{-1}G_p^T(\Lambda_1+\Lambda_1^T) \tag{38}$$

$$D_p = -R_p^{-1}G_p^T(\Lambda_2+\Lambda_2^T) \ . \tag{39}$$

Equation (37) directly gives:

$$C_e = -R_e^{-1}G_e^T(\Lambda_1+\Lambda_1^T). \tag{40}$$

It should be noted that the solution of (35), (36), and (37) for the control gains is facilitated by the fact that the problem is "separable." That is, the equations for the pursuer's controls, (35) and (36), are independent of the parameters of the evader's system, i.e., G_e, R_e. Similarly, (37), which determines the evader's controls, is independent of G_p and R_p, the pursuer's parameters.

The Euler equations for optimality are:

$$\dot{\Lambda}_1^T = -\frac{\partial\mathcal{H}}{\partial Y} \quad , \quad \Lambda_1(t_f) = \frac{a^2}{2} I \tag{41}$$

$$\dot{\Lambda}_2^T = -\frac{\partial\mathcal{H}}{\partial P} \quad , \quad \Lambda_2(t_f) = 0 \tag{42}$$

By carrying out the operations indicated in (41) and (42), and by defining a new set of multipliers

$$\Gamma_1(t) \triangleq \left[\Lambda_1(t) + \Lambda_1^T(t)\right] \tag{43}$$

$$\Gamma_2(t) \triangleq \left[\Lambda_2(t) + \Lambda_2^T(t)\right] \tag{44}$$

and by using (38), (39), and (40), these differential equations become

$$\dot{\Gamma}_1 = \Gamma_1(G_pR_p^{-1}G_p^T - G_eR_e^{-1}G_e^T)\Gamma_1 \quad , \quad \Gamma_1(t_f) = a^2 I \tag{45}$$

$$\dot{\Gamma}_2 = (\Gamma_2+\Gamma_1)G_pR_p^{-1}G_p^T(\Gamma_2+\Gamma_1) - \Gamma_1(G_pR_p^{-1}G_p^T - G_eR_e^{-1}G_e^T)\Gamma_1$$
$$+ \Gamma_2PH^TQ^{-1}H + H^TQ^{-1}HP\Gamma_2 \quad , \quad \Gamma_2(t_f) = 0 \tag{46}$$

The solution to (47) is recognized as

$$\Gamma_1(t) = K^{-1}(t_f,t) \tag{47}$$

where $K(t_f,t)$ is given by (10). Thus, C_p and C_e are the same feedback gain matrices employed by the pursuer and the evader respectively in the deterministic problem.

Employing (47) in (46), the differential equation for Γ_2 becomes

$$\dot{\Gamma}_2 = \Gamma_2 G_p R_p^{-1} G^T \Gamma_2 + \Gamma_2 \left[G_p R_p^{-1} G_p^T K^{-1} + P H^T Q^{-1} H \right]$$
$$+ \left[K^{-1} G_p R_p^{-1} G_p^T + H^T Q^{-1} H P \right] \Gamma_2 + K^{-1} G_e R_e^{-1} G_e^T K^{-1} \quad , \quad \Gamma_2(t_f) = 0 \tag{48}$$

Thus the optimal feedback strategies are given by

$$u^o(t) = -R_p(t) G_p^T(t) K^{-1}(t_f,t) y(t) - R_p(t) G_p^T(t) \Gamma_2(t) \tilde{y}(t) \tag{49}$$

$$v^o(t) = -R_e(t) G_e^T(t) K^{-1}(t_f,t) \hat{y}(t) \tag{50}$$

Note that since $R_e(t)$ is positive definite the inverse of C_e does exist if conditions i. and ii. are met. Therefore, these two conditions are sufficient for the pursuer to be able to determine the error in the evader's estimate.

The feedback strategies (49) and (50) were derived on the basis that these strategies were of the form noted in (22) and (24). To prove that these strategies are indeed optimal against all other possible forms, it must be shown that they satisfy the saddle point condition (21). This has been accomplished by first assuming that $v(t)$ is given by (50) and determining the optimal control for the pursuer based on this information. The second step is to assume that $u(t)$ is given by (49) with the pursuer obtaining a $\tilde{y}(t)$, from whatever $v(t)$ may be employed, by

$$\tilde{y}(t) = y(t) + \left[R_e^{-1} G_e^T K^{-1} \right]^{-1} v(t) \tag{51}$$

and then to determine the optimal control for the evader based on this information. The details of this verification are shown in [2].

DISCUSSION

The "Certainty-Equivalence Principle"

The optimal feedback strategies as given in (49) and (50) indicate the applicability of the "certainty-equivalence principle" of optimal control theory to the stochastic differential game problem. (See Tou and Joseph, 1961 [3], and Franklin and Gunckel, 1962 [4].) From (50) it can be seen that the evader merely uses the feedback strategy employed in the deterministic game to operate on its optimal estimate of the state $y(t)$.

From the point of view of the pursuer, the optimal strategy consists of the same feedback control used in the deterministic tame, plus an additional term to take advantage of the inaccuracy of the evader's knowledge of the state. From (49), it can be noted that whenever the evader's estimate of the state is exact, the pursuer's control is exactly the same as in the deterministic case. Though the pursuer knows that the evader will deviate from the deterministic strategy in the future—because of the noise of its measurements—the pursuer cannot utilize this information.

The Classical Interception Problem

The classical interception problem in euclidean space, with the evader making noisy measurements, can be formulated so as to be solved by the above analysis. For each euclidean dimension there is one control, and although there are two state variables, only one is "interesting" since interception, not rendezvous, is the objective; consequently condition i. is satisfied. This problem is considered in depth in [2]

CONCLUSIONS AND FURTHER WORK

The problem solved in this paper (a more detailed analysis is found in [2]) indicates that the well known "certainty-equivalence principle" of control theory is applicable to a class of stochastic differential games. This class, however, limits the dimension of the reduced state vector, i.e., the number of state variables which are to be pursued or evaded.

It is interesting to note how the analysis would proceed if this restriction is lifted. The first step would be to assume that the evader's controller is given as in (50). From control theory it could then be determined that the pursuer would employ a strategy similar to the one in (49). However, because the inverse of $G(t_f,t)$ would not exist, the pursuer would have to estimate $\tilde{y}(t)$ and use this estimate in lieu of the actual value.

Again, however, the pursuer's estimate of $\tilde{y}(t)$ would not be perfect, and the evader could take advantage of this imperfection. Again using control theory, the knowledge of the inaccuracies of the pursuer's estimate of $\tilde{y}(t)$ and the procedure for obtaining this estimate, the evader would obtain a feedback strategy based on a) the evader's estimate of the state, $y(t)$, and b) the evader's estimate of the pursuer's estimate of $\tilde{y}(t)$. Again, this latter estimate would not be perfect.

The difficulty that is encountered is now obvious; there appears to be an infinite number of terms in the optimal strategies of each of the two players. These terms are each based on estimates of estimates of estimates of... . We shall call this problem the "closure" problem in stochastic pursuit-evasion games. With assumptions i. and ii., closure is possible.

The authors are presently working to resolve this difficulty. The obvious conjecture here is that these estimates of estimates form an asymptotic series which can be truncated for practical purposes.

REFERENCES

The work presented here was supported by the Nonr Contract 1866(16).

[1] Y. C. Ho, A. E. Bryson, Jr., and S. Baron, "Differential Games and Optimal Pursuit-Evasion Strategies," IEEE Trnasactions on Automatic Control, AC-10, No. 4, October 1965, pp. 385-389.

[2] R. D. Behn and Y. C. Ho, "On A Class of Linear Stochastic Differential Games, TR No. 542, Division of Engineering and Applied Physics, Harvard University, October 1967.

[3] P. Joseph and J. Tou, "On Linear Control Theory," Transactions of the AIEE, Part III, September 1961.

[4] T. F. Gunckel and G. F. Franklin, "A General Solution for Linear Sampled-Data Control Systems," Journal of Basic Engineering, Vol. 85, Series D, No. 2, Transactions of the ASME, 1963, pp. 197-203.

PERFORMANCE MOMENT RECURSIONS, WITH APPLICATION TO EQUALIZER CONTROL LAWS

MICHAEL K. SAIN
University of Notre Dame
Notre Dame, Indiana

ABSTRACT

Optimal control of a Langevin-type plant, in which random internal disturbances are present, often leads to a random performance index. The main result shows that the moments of such an index can be calculated recursively by means of the same general class of procedures which are commonly used to determine controllers minimizing the average cost of operating the plant. In principle, therefore, it is possible to study the performance index as a random variable, to the order desired, simply by extending existing first order methods recursively. In some cases, for which solutions of the minimum average cost problem are available in closed form, it is possible to establish the moment recursions also in closed form. The results, in a special case, apply also to the study of the existence and properties of equalizer control laws, which are thus characterized by a quite general set of conditions.

INTRODUCTION

This work further explores the properties of the probability distribution of performance in control problems over Markov processes. In order to be more explicit, let the controlled process x(t) be described by the stochastic differential equation

$$\dot{x}(t) = f(t,x(t),k(t,x(t))) + w(t), \quad t_0 \leq t \leq t_F, \tag{1}$$

where w(t) is an n-dimensional, continuous parameter, Gaussian white noise process. The control law k is a function such that $T \times R^n \rightarrow R^m$, where $T = [t_0,t_F]$ and the m-fold Cartesian product of the real line R is denoted by R^m. Corresponding to different control laws, (1) generates[1] processes under a given initial value $x(t_0) = x_0$. The control law k represents a feedback strategy which determines the control action u(t) according to the rule

$$u(t) = k(t,x(t)). \tag{2}$$

In order to specify a control problem over the process (1), it is necessary to be able to compare various control laws with respect to an objective. The choice of an objective is often described relative to a single random variable J(t,x:k) defined over the process and control

---

[1] The precise meaning of (1) is usually inferred from the integral equation

$$x(t) - x_0 = \int_{t_0}^{t} f(s,x(s),k(s,x(s)))ds + \int_{t_0}^{t} dw(s)$$

where w(t) is the Wiener process which has unbounded variation with probability one. It is therefore necessary to introduce the Wiener integral in order to interpret (1). It is not necessary, however, to introduce the stochastic integral of Ito. For a much more comprehensive discussion of these points, see for example the work of Wong and Zakai [1].

action segments $x(s)$ and $u(s)$, $s \in T$, in the manner

$$J(t,x:k) = \int_{t}^{t_F} L(s,x(s),u(s))ds \tag{3}$$

where L is a mapping from $T \times R^n \times R^m$ to the nonnegative real line. For example, let

$$V_1(t,x:k) = E\left\{J(t,x:k) \mid x(t) = x\right\} ; \tag{4}$$

an often chosen objective relative to (4) is to find k_1° such that

$$V_1(t,x:k_1^\circ) = V_1^\circ(t,x) \leq V_1(t,x:k) \tag{5}$$

whenever $k_1^\circ \neq k$. In (5), and throughout the remainder of the paper, it is implicitly assumed that $x(t)$ and $u(t)$ may take on any finite values in R^n and R^m respectively. Introductory treatment of problems in this category has been given by Wonham [2]. As a second example, let

$$V_1(t,x:k) = V_1(t,x) \tag{6}$$

where $V_1(t,x)$ is a given constraint function. Further, define

$$V_2(t,x:k) = E\left\{J^2(t,x:k) \mid x(t) = x\right\} . \tag{7}$$

Another class of control problems is then associated with finding k_2° such that

$$V_2(t,x:k_2^\circ) = V_2^\circ(t,x) \leq V_2(t,x:k) \tag{8}$$

whenever $k_2^\circ \neq k$. Both k_2° and k in (8) must satisfy (6), and (6) itself must also be compatible with (5). Ongoing investigation of this type of problem is being carried out by the author [3].

The present results are explicitly concerned with the properties of the functions

$$V_i(t,x:k) = E\left\{J^i(t,x:k) \mid x(t) = x\right\},\ i = 0, 1, 2, \ldots, \tag{9}$$

which, in the aggregate, are characteristic of the conditional probability density $p_J(\cdot \mid x(t) = x)$ of (3). It is shown that these moment functions satisfy recursive differential equations, which in certain cases can be solved by a particularly simple iteration. The results point the way toward a straightforward computational procedure for studying higher order moments in stochastic optimum control problems. Further, they have a potential for analytical application as well. As an example, they are here applied to the problem of characterizing some of the properties of equalizer control laws.

PROPERTIES OF MARKOV PROCESSES

Corresponding to (1), it is possible to define a transition probability distribution.[2]

---

[2] The properties summarized briefly here are outlined in somewhat greater detail in the survey article by Wonham [2]. Where possible, the notation used in these remarks is intended to conform with his usage.

Definition 1. The transition probability distribution under k, denoted in the manner P(t,x:s,E:k), is defined by

$$P(t,x:s,E:k) = \mathrm{Prob}[x(s) \in E \mid x(t) = x;\ k],\quad s \geq t \geq t_0, \tag{10}$$

where $E \subset R^n$ and

$$\lim_{t \to s} P(t,x:s,E:k) = 1,\ x \in E \tag{11A}$$

$$= 0,\ x \notin E\ . \tag{11B}$$

If the corresponding density function p(t,x:s,y:k) exists, then

$$\int_E p(t,x:s,y:k)dy = P(t,x:s,E:k) \tag{12A}$$

and

$$\lim_{s \to t} p(t,x:s,y:k) = \delta(x - y), \tag{12B}$$

where $\delta(\cdot)$ is the Dirac delta function.

A fact of considerable utility in the sequel is that the transition functions (11) and (12) satisfy Kolmogorov's backward equation. The equation is stated in Lemma 1 below. It is convenient first, however, to introduce the following three definitions.

Definition 2. Let p be a function of x from R^n to the nonnegative real[3] line. Then $\partial p/\partial x$ is a column vector such that

$$(\partial p/\partial x)_i = \partial p/\partial x_i,\ i = 1, 2, \ldots, n, \tag{13A}$$

and $\partial^2 p/\partial x^2$ is an n x n matrix such that

$$(\partial^2 p/\partial x^2)_{ij} = \partial^2 p/\partial x_i \partial x_j,\ i,j = 1, 2, \ldots, n. \tag{13B}$$

Definition 3. Let A and B be n x n matrices. Then

$$A \circ B = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} B_{ij}\ , \tag{14}$$

where A_{ij} is the ij th element of A.

Definition 4. In (1)[4], let the (zero mean) process w(t) have a covariance matrix Q(t), i.e.,

$$E\left\{w(t)w'(s)\right\} = Q(t)\ \delta(t - s), \tag{15}$$

where the superscript (′) denotes transposition; then $O_{x,t}(Q,f)$ is Kolmogorov's backward operator defined by

$$O_{x,t}(Q,f) = (\partial/\partial t) + \tfrac{1}{2}Q \circ (\partial^2/\partial x^2) + f'(\partial/\partial x) \tag{16}$$

where arguments are omitted for brevity.

---

[3] Other arguments of p have been suppressed for simplicity.

[4] Equation (1) is often termed a Langevin equation. A number of non-Markovian processes can be converted to the Langevin form by augmenting the original state vector.

With the completion of these preliminaries, and under suitable differentiability conditions, it is possible to state the well known

Lemma 1. The transition probability distribution (10) and the transition probability density (12), corresponding to the stochastic differential equation (1), satisfy (respectively) the equations

$$O_{x,t}(Q,f)\ P(t,x:s,E:k) = 0 \tag{17A}$$

$$O_{x,t}(Q,f)\ p(t,x:s,y:k) = 0 \tag{17B}$$

subject to the boundary conditions (11) and (12B).

A partial differential equation of type (17) will be called a Kolmogorov differential equation; such equations may be homogeneous, as (17) above, or inhomogeneous, as (18) below.

RECURSIVE GENERATION OF PERFORMANCE MOMENTS

Consider once again the first performance moment, $V_1(t,x:k)$, which was briefly discussed in the Introduction. It is a comparatively well known fact that this function satisfies a Kolmogorov differential equation, which is re-stated in Lemma 2 to mark the first step in what proves to be a general recursion.

Lemma 2. $V_1(t,x:k)$, the conditional average cost function, satisfies the nonhomogeneous Kolmogorov differential equation

$$O_{x,t}(Q,f)\ V_1(t,x:k) + L(t,x,k) = 0 \tag{18}$$

together with the boundary condition $V_1(t_F,x:k) = 0$.[5]

Proof. Lemma 2, or its equivalent, may be found in divers forms in the literature, as for example in [2, p. 117]. A relatively direct way to verify (18) is to observe that (17B) may be multiplied in both members by $L(s,y,k)$ and integrated over $t < s < t_F$ and $y \in R^n$, provided that the impulsive boundary condition (12B) is taken into account.

Since $V_0(t,x:k) = 1$, it is possible to view (18) as an equation of the form

$$O_{x,t}(Q,f)\ V_{i+1}(t,x:k) + (i+1)\ L(t,x,k)\ V_i(t,x:k) = 0 \tag{18$'$}$$

for the case $i = 0$. It is the fundamental result of this paper that the equation (18′) is valid also for $i = 1, 2, \ldots$, as expressed in the following theorem.

Theorem 1. $V_{i+1}(t,x:k)$, the condition average of the $(i+1)^{st}$ power of the performance index (3), is related to $V_i(t,x:k)$, the conditional average of the i^{th} power of (3), by the differential equation

$$O_{x,t}(Q,f)\ V_{i+1}(t,x:k) + (i+1)\ L(t,x,k)\ V_i(t,x:k) = 0\ , \tag{19}$$

together with the boundary condition $V_{i+1}(t_F,x:k) = 0$,[5] $i = 0, 1, 2, \ldots.$

---

[5] The problem considered here may be generalized, if the situation warrants, by addition of a terminal cost function $J_F(x(t_F))$ to (3). In this case, the boundary conditions associated with (18) and (19) are the first and $(i+1)^{st}$ powers of $J_F(x)$, respectively. If $k = k_1^\circ$, the Kolmogorov differential equation (18) is sometimes called a stochastic Hamilton-Jacobi equation, even though it is not stochastic and also contains second order derivatives. The usage is by analogy with the deterministic case.

Proof. Theorem 1 is established by generalizing the approach of Lemma 2. Corresponding to (10), define the joint conditional probability distribution

$$P(t,x:s_1,E_1:s_2,E_2:\ldots:s_{i+1},E_{i+1}:k) =$$
$$\mathrm{Prob}[x(s_1)\in E_1, x(s_2)\in E_2, \ldots, x(s_{i+1})\in E_{i+1} \mid x(t) = x;\ k],$$
$$s_i > t > t_0,\ E_i \subset R^n,\ i = 1,\ 2,\ 3,\ \ldots \tag{20A}$$

and identify the joint conditional probability density function $p(t,x:s_1,y_1:\ldots:s_{i+1},y_{i+1}:k)$ corresponding to (20A) by extending the integral over E in (12A) to an (i+1)fold integral over $E_1,E_2,\ldots,E_{i+1}$. By the chain rule, it is always possible to write this conditional density function as a product

$$p(t,x:s_j,y_j:k)p(s_j,y_j:s_1,y_1:\ldots:s_{j-1},y_{j-1}:s_{j+1},y_{j+1}:\ldots:s_{i+1},y_{i+1}:k)$$

where the x and t dependence is isolated in the first factor by reason of the Markovian nature of (1) and j is the integer such that $|s_i-t|$ is a minimum, i = 1, 2, ..., i+1. From this expression it follows that

$$O_{x,t}(Q,f)\ p(t,x:s_1,y_1:\ldots:s_{i+1},y_{i+1}:k) = 0 \tag{20B}$$

and

$$\lim_{s_j\to t}\ p(t,x:s_1,y_1:\ldots:s_{i+1},y_{i+1}:k) = \tag{20C}$$
$$= \delta(y_j-x)\ p(t,x:s_1,y_1:\ldots:s_{j-1},y_{j-1}:s_{j+1},y_{j+1}:\ldots:s_{i+1},y_{i+1}:k).$$

The relationship (19) can then be obtained by multiplying both members of (20B) by

$$L(s_1,y_1,k(s_1,y_1))L(s_2,y_2,k(s_2,y_2))\cdots L(s_{i+1},y_{i+1},k(s_{i+1},y_{i+1})) \tag{20D}$$

and integrating over $E_1,E_2,\ldots E_{i+1}$. Since $O_{x,t}(Q,f)$ commutes with these integrations, the result is the expression

$$O_{x,t}(Q,f)\ E\left\{L_1(s_1)L_2(s_2)\cdots L_{i+1}(s_{i+1}) \mid x(t) = x;\ k\right\} = 0 \tag{20E}$$

where $L_j(s_j) = L(s_j,y_j,k(s_j,y_j))$, which can then be integrated i+1 times over $t < s_i < t_F$, i = 1, 2, ..., i+1. Finally, $O_{x,t}(Q,f)$ can be interchanged with respect to these last integrations, provided that a compensating term is added to account for the dependence of the lower integration limit on t. This compensating term, by inspection and application of (20C), is simply

$$(i+1)\ L(t,x,k)\ V_i(t,x:k) \tag{21}$$

and the net result is (19).

Theorem 1 can be applied in a number of ways to shed further light on stochastic control problems. In the remainder of the paper, two such applications are briefly discussed: the study of *equalizer control laws* and the generation of higher cost moments in linear stochastic regulator problems.

APPLICATION TO EQUALIZER CONTROL LAWS

Sworder [4] has defined and studied some properties of equalizer control laws. Herein it is sufficient to characterize such a control law as follows.

Definition 5. A control law k is an equalizer control law, denoted k_e, if and only if

$$E\left\{J^2(t,x:k) - V_1^2(t,x:k) \mid x(t) = x;\ k\right\} = 0, \tag{22}$$

that is, if and only if $J(t,x:k) = V_1(t,x:k)$ in the mean square sense. In addition to their academic interest, equalizer control laws appear to have a real, practical significance in the construction of other, more sophisticated strategies. For discrete-time systems over the real field, Sworder [4] has suggested that an effective means for constructing minmax control laws is to begin with equalizer laws.

With this motivation, it seems worthwhile to characterize the properties of equalizer control laws for continuous time systems.[6] The following lemma depends directly on Definition 5 and (9).

Lemma 3. A control law k is an equalizer control law if and only if

$$V_2(t,x:k) = V_1^2(t,x:k). \tag{23}$$

Now from Theorem 1 and Lemma 3 it follows that the function $V_1(t,x:k_e)$ must satisfy two simultaneous equations.

Theorem 2. $V_1(t,x:k_e)$, the performance function corresponding to the equalizer control law k_e, satisfies the simultaneous equations

$$O_{x,t}(Q,f_e)\, V_1(t,x:k_e) + L(t,x,k_e) = 0 \tag{24A}$$

$$\|\partial V_1(t,x:k_e)/\partial x\|^2_{Q(t)} = 0 \tag{24B}$$

where the subscript e on f signifies that k is replaced by k_e and the notation $\|a\|^2_A$ denotes $a'Aa$.

Proof. The first of the above relationships, namely (24A), is just a statement of Lemma 2 for the control law k_e. It remains, therefore, to establish (24B). From Theorem 1, i = 1, write

$$O_{x,t}(Q,f_e)\, V_2(t,x:k_e) + 2L(t,x,k_e)\, V_1(t,x:k_e) = 0 \tag{25}$$

which by Lemma 3 becomes

$$O_{x,t}(Q,f_e)\, V_1^2(t,x:k_e) + 2L(t,x,k_e)\, V_1(t,x:k_e) = 0. \tag{26}$$

It is then convenient to identify $O_{x,t}(Q,f_e)$ as the sum

$$O_{x,t}(Q,f_e) = O^1_{x,t}(f_e) + O^2_x(Q) \tag{27}$$

of two operators, the first of which consists only of first partial derivatives

$$O^1_{x,t}(f_e) = (\partial/\partial t) + f_e'(\partial/\partial x) \tag{28}$$

---

[6] It is not intended here to exhaustively study the existence of equalizer control laws, but rather to provide a structure in which that existence may subsequently be better examined.

and the second of which consists of second partial derivatives

$$O_x^2(Q) = (1/2)Q\circ(\partial^2/\partial x^2). \qquad (29)$$

With these definitions, (26) becomes

$$O_{x,t}^1(f_e)\, V_1^2(t,x:k_e) + O_x^2(Q)\, V_1^2(t,x:k_e) + $$
$$+ 2L(t,x,k_e)\, V_1(t,x:k_e) = 0. \qquad (30)$$

But

$$O_{x,t}^1(f_e)\, V_1^2(t,x:k_e) = 2V_1(t,x:k_e)\, O_{x,t}^1(f_e)\, V_1(t,x:k_e), \qquad (31)$$

so that (30) becomes, when combined with (24A),

$$O_x^2(Q)\, V_1^2(t,x:k_e) - 2V_1(t,x:k_e)\, O_x^2(Q)\, V_1(t,x:k_e) = 0, \qquad (32)$$

which can be rewritten then to give

$$(1/2)Q\circ[(\partial^2 V_1^2(t,x:k_e)/\partial x^2) - 2V_1(t,x:k_e)(\partial^2 V_1(t,x:k_e)/\partial x^2)]$$
$$= 0. \qquad (33)$$

Finally, (33) and (14) imply

$$||(\partial V_1(t,x:k_e)/\partial x)||_{Q(t)}^2 = 0 \qquad (34)$$

as desired.

Theorem 2 implies a number of additional results. Some of these will be written in a later section. An important special case, however, arises when $Q(t)$ is positive definite.[7]

Corollary 2.1. Let k_e be an equalizer control law and let $Q(t)$ be a positive definite matrix; then $L(t,x,k_e)$ must be identically zero.

Proof. From (34),

$$(\partial V_1(t,x:k_e)/\partial x) = 0, \qquad (35)$$

so that $V_1(t,x:k_e)$ is essentially a function of t. But (24A), which in this case becomes

$$(dV_1(t,x:k_e)/dt) + L(t,x,k_e) = 0, \qquad (36)$$

must necessarily be satisfied for all $x \in R^n$. Such a situation implies that L is essentially a function of t, which represents an improperly posed problem, since the terminal time t_F has been assumed a given quantity. The only remaining possibility is that L vanishes identically.[8] The implication of this corollary is that the search for equalizer control laws should be directed to the case in which $Q(t)$ is not positive definite.

---

[7] A symmetric matrix A is said to be positive definite if $a'Aa > 0$ whenever $a \neq 0$.

[8] The conclusion is not inconsistent with the work of Sworder [4], since only terminal payoff functions were considered in that formulation. If Q(t) is assumed to be positive definite, then the results herein may be reconsidered for the case L = 0.

APPLICATION TO LINEAR STOCHASTIC REGULATORS

Consider now the case in which the cost function $V_1(t,x{:}k_e)$ is quadratic in the manner

$$V_1(t,x{:}k_e) = x'Z_{1e}(t)x + z_{1e}(t). \tag{37}$$

From Theorem 2, it is necessary that (24) be satisfied, which gives the pair of equations

$$Q(t)\circ Z_{1e}(t) + \dot{z}_{1e}(t) = 0,\ z_{1e}(t_F) = 0 \tag{38A}$$

$$x'\dot{Z}_{1e}(t)x + 2f_e'Z_{1e}(t)x + L(t,x,k_e) = 0,\ Z_{1e}(t_F) = 0\ ; \tag{38B}$$

$$Z_{1e}(t)\ Q(t)\ Z_{1e}(t) = 0. \tag{39}$$

It is of course explicitly assumed that $Z_{1e}(t)$ is a symmetric matrix; it is further assumed, implicitly in (38), that there is no t dependence in L which can be additively separated from the x dependence in L.

If it is further assumed that the underlying system (1) is linear in x and k, and that the control law k is linear in x, according to the relationships

$$f_e(t,x,k_e) = A(t)x + B(t)k_e \tag{40}$$

$$k_e(t,x) = K_e(t)x, \tag{41}$$

and that the integrand of the cost (3) is quadratic in the manner

$$L(t,x,k_e) = (1/2)x'L_x(t)x + (1/2)k_e'L_u(t)k_e, \tag{42}$$

then (38B) becomes

$$\dot{Z}_{1e}(t) + A_e'(t)Z_{1e}(t) + Z_{1e}(t)A_e(t) + (1/2)[L_x(t) + K_e'(t)L_u(t)K_e(t)] = 0, \tag{43}$$

where $A_e(t)$ is the closed loop state matrix

$$A_e(t) = A(t) + B(t)K_e(t). \tag{44}$$

The study of equalizer control laws in this case then reduces to the search for matrices $K_e(t)$ such that the solution to (43), under the boundary condition in (38B), satisfies (39). It is clear from Corollary 2.1 that no such control law exists for the first order system n = 1. Finally, it should be noted that (39) demands that $Z_{1e}(t)$ be singular.[9] If such were not the case, then Q(t) could only be the trivial zero matrix.

---

[9] An interesting exercise is to compare (39) with the condition $Q(t)\circ Z_{1e}(t) = 0$, since by (38A) this would imply $z_{1e}(t) = 0$, which might be taken as an alternate characterization of the equalizer law. The singularity condition on $Z_{1e}(t)$ also suggests that the noise-free system ($w(t) = 0$) may have to satisfy unobservability conditions; for it is known, at least for control laws which are optimal in the sense of minimizing $V_1(t,x{:}k)$, that observability leads to nonsingularity of $Z_{1e}(t)$. See [5]. These remarks are closely related to invariance theory [6].

In the remaining paragraphs, the recursion outlined in Theorem 1 is considered for the special case of linear stochastic regulators. Because of the space limitation, only a first order example will be developed. The notation needed is just that of (37), (38), (40), (41), (42), (43), and (44) with the subscript e removed wherever it appears in those expressions.

By reason of the assumptions on w(t), and under the hypotheses of linear system, linear control law, and quadratic cost, it can be simply established by construction that the form of the i^{th} moment $V_i(t,x:k)$ must be

$$V_i(t,x:k) = Z_{i,2i}(t)x^{2i} + Z_{i,2i-2}(t)x^{2i-2} + \dots + \\ + Z_{i,2}(t)x^2 + Z_{i,0}(t) . \qquad (45)$$

Equation (19) becomes

$$\dot{Z}_{i+1,2i+2}(t)x^{2i+2} + \dot{Z}_{i+1,2i}(t)x^{2i} + \dots + \dot{Z}_{i+1,2}(t)x^2 + \dot{Z}_{i+1,0}(t) + \\ + [A(t) + B(t)K(t)][(2i+2)Z_{i+1,2i+2}(t)x^{2i+2} + 2iZ_{i+1,2i}(t)x^{2i} + \\ + \dots + 2Z_{i+1,2}(t)x^2] + (1/2)Q(t)[(2i+2)(2i+1)Z_{i+1,2i+2}(t)x^{2i} + \\ + (2i)(2i-1)Z_{i+1,2i}(t)x^{2i-2} + \dots + (4)(3)Z_{i+1,4}(t)x^2 + \\ + (2)(1)Z_{i+1,2}(t)] + (1/2)(i+1)[L_x(t) + L_u(t)K^2(t)][Z_{i,2i}(t)x^{2i+2} + \\ + Z_{i,2i-2}(t)x^{2i} + \dots + Z_{i,2}(t)x^4 + Z_{i,0}(t)x^2] = 0. \qquad (46)$$

Examination of (46) gives the explicit recursions

$$\dot{Z}_{i+1,2i+2}(t) + (2i+2)[A(t)+B(t)K(t)]Z_{i+1,2i+2}(t) + \qquad (47) \\ + (i+1)(1/2)[L_x(t)+L_u(t)K^2(t)]Z_{i,2i}(t) = 0,\ Z_{i+1,2i+2}(t_F) = 0;$$

$$\dot{Z}_{i+1,2i}(t) + (2i)[A(t)+B(t)K(t)]Z_{i+1,2i}(t) + (1/2)Q(t)(2i+2)(2i+1)\cdot \\ \cdot Z_{i+1,2i+2}(t) + (1/2)(i+1)[L_x(t)+L_u(t)K^2(t)]Z_{i,2i-2}(t) = 0, \\ Z_{i+1,2i}(t_F) = 0; \qquad (48)$$

.....

$$\dot{Z}_{i+1,2}(t) + (2)[A(t)+B(t)K(t)]Z_{i+1,2}(t) + (1/2)Q(t)(4)(3)Z_{i+1,4}(t) + \\ + (1/2)(i+1)[L_x(t)+L_u(t)K^2(t)]Z_{i,0}(t) = 0, \\ Z_{i+1,2}(t_F) = 0; \qquad (49)$$

$$\dot{Z}_{i+1,0}(t) + (1/2)Q(t)(2)(1)Z_{i+1,2}(t) = 0,\ Z_{i+1,0}(t_F) = 0. \qquad (50)$$

As an example, let $A(t)+B(t)K(t) = 1$, $L_x(t)+L_u(t)K^2(t) = Q(t) = 2$. Then

$$Z_{0,0}(t) = 1; \qquad (51)$$

$$Z_{1,2}(t) = \int_t^{t_F}[\exp-2(t-s)]ds = (1/2)[e^{2(t_F-t)} - 1]; \qquad (52)$$

$$Z_{1,0}(t) = \int_t^{t_F} 2Z_{1,2}(s)ds = (1/2)[e^{2(t_F-t)} - 1] - (t_F - t); \qquad (53)$$

$$Z_{2,4}(t) = \int_t^{t_F} 2[\exp-4(t-s)]Z_{1,2}(s)ds$$
$$= (1/4)[e^{2(t_F-t)} - 1]^2; \tag{54}$$

$$Z_{2,2}(t) = \int_t^{t_F} [\exp-2(t-s)][12Z_{2,4}(s) + 2Z_{1,0}(s)]ds; \tag{55}$$

$$Z_{2,0}(t) = \int_t^{t_F} 2Z_{2,2}(s)ds; \tag{56}$$

....

Among the expressions above, (55) is quite typical since it points out the dependence of the recursion on the next highest coefficient in the present series and the next lowest coefficient in the previous series. The calculations above, though perhaps tedious, do indicate a methodical and straightforward method for studying the moments of the distribution of $J(t,x{:}k)$.

CONCLUSION

The main result of the paper is Theorem 1, which outlines a recursive procedure for methodically studying the conditional moments of the cost functional in stochastic optimal control problems over Markov processes. The techniques are straightforward and can be terminated at any desired order moment. The distribution function of the cost is not directly computed, but is indirectly implied by the set of moments. The results can also be applied to deduce properties of certain types of control schemes, such as the equalizer control law.

ACKNOWLEDGMENT

The results reported herein were supported by the National Science Foundation under Grant GK 804.

REFERENCES

1. E. Wong and M. Zakai, "On the relation between ordinary and stochastic differential equations," Int. J. Engng. Sci., vol. 3, pp. 213-229, 1965.
2. W. M. Wonham, "Stochastic problems in optimal control," IEEE Int. Conv. Rec., part 2, pp. 114-124, 1963.
3. M. K. Sain, "A sufficiency condition for minimum variance control of markov processes," Proc. Fourth Allerton Conf. Circuit System Theory, pp. 593-599, 1966.
4. D. Sworder, Optimal Adaptive Control Systems. New York: Academic Press, 1966.
5. R. E. Kalman, T. S. Englar, and R. S. Bucy, "Fundamental study of adaptive control systems," Tech. Rept. ASD-TR-61-27, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio, 1962.
6. J. B. Cruz, Jr., and W. R. Perkins, "Conditions for signal and parameter invariance in dynamical systems," IEEE Trans., vol. AC-11, pp. 614-615, 1966.

LEARNING IN NONSTATIONARY ENVIRONMENT USING DYNAMIC STOCHASTIC APPROXIMATION

Y. T. CHIEN
University of Connecticut
Storrs, Connecticut

K. S. FU
Purdue University
Lafayette, Indiana

ABSTRACT

The problem of learning in nonstationary environment is formulated as that of estimating time-varying parameters of a probability distribution which characterizes the process under study. Dynamic stochastic approximation procedures are proposed to estimate the unknown time-varying parameters. Both supervised and nonsupervised learning schemes are discussed and convergence properties are investigated. An accelerated scheme for the possible improvement of the dynamic algorithm is given.

I. INTRODUCTION

This paper is intended to be a continued effort in treating learning systems via the method of stochastic approximation [4]. The main problem considered here is the learning of time-varying parameters in a random environment characterized by some probability distribution. Our approach is essentially an extension of the dynamic stochastic approximation procedure proposed by Dupač [3]. Both supervised and nonsupervised learning problems involving time-varying parameters are investigated by using this approach.

For a stationary random environment, it is well known that learning the parameters of a probability distribution can be treated as the successive estimation of conditional distributions through the optimal use of learning samples [1,2]. Recursive algorithms of estimates may be easily obtained when the a prori probability distribution of the true parameter can be properly chosen. Further investigation [4] has shown that these recursive algorithms may be viewed and treated within the framework of the stochastic approximation method generalized by Dvoretzky [7]. When the randon environment is nonstationary, however, these algorithms will not be able to produce the true parameters, since the parameters themselves vary with time as the learning samples are acquired. The purpose of this paper, therefore, is to establish stochastic approximation algorithms which will represent a first step toward the learning of parameters in the presence of a time-varying trend.

We shall only discuss situations where the parameters of a probability distribution are slowly time-varying in a linear or nearly-linear fashion. In general, the proposed learning algorithm will consist of a two-step approximation procedure to be performed at each stage of the learning process. The first step is designed to correct the time-varying trend of the parameter being learned; the second step is made by means of an ordinary stochastic approximation procedure, based on the observation of a new learning sample. To justify the use of this algorithm, convergence properties of the successive estimates are investigated. By using Dvoretzky's approach, it is shown that the estimates resulting from the algorithm converge to the true parameter value in mean square and with probability one. Furthermore, expression of mean square errors of successive estimates is derived, from which optimal learning sequence can be constructed through a standard minimization process. It is found that the speed of convergence in terms of mean square error, although somewhat slower, is in the same order as that of the algorithm for learning in stationary environment [4]. A practical method of improving the convergence speed is suggested and a numerical example is presented to illustrate the effectiveness of the possible improvement.

II. SUPERVISED LEARNING IN NONSTATIONARY ENVIRONMENT

2.1 The Learning Algorithm

The problem of supervised learning may be considered to have the following situation: We have a sequence of samples, $Y_1, Y_2, \ldots, Y_n$, known to be generated from a conditional probability density $p(Y/\theta)$, where θ is a set of unknown parameters (e.g., mean value, variance, etc.) characterizing the probability density (or probability distribution). The samples $Y_1, Y_2, \ldots, Y_n$ thus generated will be called classified learning samples. Usually the conditional probability density represents a random environment whose uncertainty is reflected in the set of parameters θ. When the random environment is nonstationary (time-varying) as in the case we now consider, θ becomes a function of the time index n. Let this set of time-varying parameters be denoted by θ_n. Our concern here is to construct a sequence of estimates which will in some sense best approximate, and asymptotically converge to, the true parameter set θ_n as the learning samples $Y_1, Y_2, \ldots, Y_n$ are acquired.

Let us first consider the case where the parameter set θ_n is the unknown mean value of $p(Y/\theta_n)$ with known variance σ_n^2. The parameter varies slowly with time according to the following situation:

$$\theta_{n+1} = (1 + n^{-1})\,\theta_n + O(n^{-\mu}), \quad \mu \geqq 1 \tag{1}$$

We denote the nth estimate of θ_n by X_n. Then a two-step stochastic approximation algorithm is defined as follows:

$$\text{First step:} \quad X_n^* = (1 + n^{-1})\,X_n + O(n^{-\mu}) \tag{2}$$

$$\text{Second step:} \quad X_{n+1} = X_n^* + \gamma_n\,(Y_{n+1} - X_n^*) \tag{3}$$

where γ_n is a sequence of positive numbers satisfying the conditions:

$$\lim_{n\to\infty} \gamma_n = 0, \quad \sum_{n=1}^{\infty} \gamma_n = \infty \quad \text{and} \quad \sum_{n=1}^{\infty} \gamma_n^2 < \infty \tag{4}$$

and Y_{n+1} is the classified learning sample received at the (n+1)th stage.

The meaning of the algorithm described in (2) and (3) is the following: At the time instant n+1, we try to obtain an estimate X_{n+1} for the value of θ_{n+1}. The first step is to make a correction for the time-varying trend and the second step is essentially to apply an ordinary stochastic approximation procedure by means of the current learning sample Y_{n+1}. The correction term in (3) for the successive estimates is controlled by a sequence of numbers γ_n satisfying the conditions in (4).

In order to show that the use of this algorithm is justified, we shall first investigate its asymptotic properties when the number of samples becomes infinitely large. By making use of Dvoretzky's approach [7], we demonstrate that the estimate X_{n+1} converges to θ_{n+1} in mean square and with probability one. Re-writing (3), we have

$$\begin{aligned} X_{n+1} &= (1 - \gamma_n)\,X_n^* + \gamma_n\,Y_{n+1} \\ &= (1 - \gamma_n)(1 + n^{-1})(X_n + O'(n^{-\mu})) + \gamma_n\theta_{n+1} + \gamma_n\eta_{n+1} \end{aligned} \tag{5}$$

where $O'(n^{-\mu}) = (1 + n^{-1})^{-1}\, O(n^{-\mu})$ (6)

and Y_{n+1} has been replaced by $\theta_{n+1} + \eta_{n+1}$. η_{n+1} represents the noisy component of the learning sample Y_{n+1} with

$$E(\eta_n) = 0 \quad \text{and} \quad E(\eta_n^2) = \sigma_n^2. \tag{7}$$

Let $T_n(X_1, X_2, \ldots, X_n)$ denote the following transformation:

$$T_n(X_1, X_2, \ldots, X_n) = (1 - \gamma_n)(1 + n^{-1})(X_n + O'(n^{-\mu})) + \gamma_n\, \theta_{n+1} \tag{8}$$

Then $X_{n+1} = T_n(X_1, X_2, \ldots, X_n) + \gamma_n \eta_{n+1}$ (9)

which can be viewed as a stochastic approximation algorithm consisting of a deterministic (error-free) transformation T_n and a random noise component $\gamma_n\eta_{n+1}$. To ensure the convergence of X_{n+1} to θ_{n+1} in mean square and with probability one, we must show that

$$\lim_{n\to\infty} E(X_n - \theta_n)^2 = 0 \quad \text{and} \quad P(\lim_{n\to\infty} X_n = \theta_n) = 1$$

From (8) and (1), $|T_n(X_1, X_2, \ldots, X_n) - \theta_{n+1}|$

$$\begin{aligned} &= |(1 - \gamma_n)(1 + n^{-1})(X_n + O'(n^{-\mu})) + \gamma_n\theta_{n+1} - \theta_{n+1}| \\ &= (1 - \gamma_n)(1 + n^{-1})\ |X_n - \theta_n| \\ &= F_n\ |X_n - \theta_n| \end{aligned} \tag{10}$$

where $F_n = (1 - \gamma_n)(1 + n^{-1})$ (11)

Suppressing the random part of (9) and using the relation (10) recursively, we obtain

$$\begin{aligned} |X_{n+1} - \theta_{n+1}| &= F_n\ |X_n - \theta_n| \\ &= F_n \cdot F_{n-1}\ |X_{n-1} - \theta_{n-1}| \\ &= \prod_{k=1}^{n} F_k\ |X_1 - \theta_1| \end{aligned} \tag{12}$$

Thus if $\prod_{n=1}^{\infty} F_n = 0$, then X_{n+1} converges to θ_{n+1} for any finite values of θ_1 and X_1. This would be the case if we choose γ_n properly as will be shown later.

Now let $W_n = \gamma_n\eta_{n+1}$. From (7) it is obvious that $E(W_n) = 0$ and $E(W_n^2) = \gamma_n^2\sigma_{n+1}^2$. Suppose the noise component of Y_{n+1} is such that its variance σ_{n+1}^2 is bounded by σ^2 for all n. Then

$$\sum_{n=1}^{\infty} E(W_n^2) \leq \sigma^2 \sum_{n=1}^{\infty} \gamma_n^2 < \infty \tag{13}$$

which establishes the fact that the sum of the variances is finite for any possible infinite sequence. Finally, we put

$$V_n^2 = E(X_n - \theta_n)^2 \tag{14}$$

Then, after squaring and taking expectation of both sides of (5), we have the recursive relation for the mean square error of the estimates

$$V_{n+1}^2 \leq F_n^2 V_n^2 + \gamma_n^2 \sigma^2 \tag{15}$$

in which the inequality comes from the substitution of the upper bound of the noise variance σ^2 for σ_n^2. Letting $B_n^2 = \gamma_n^2\sigma^2$ and upon iterating, we have

$$\begin{aligned} V_{n+1}^2 &\leq F_n^2 V_n^2 + B_n^2 \\ &\leq F_n^2 F_{n-1}^2 \cdots F_1^2 V_1^2 + F_n^2 F_{n-1}^2 \cdots F_n^2 B_1^2 + \cdots \\ &\quad + F_n^2 F_{n-1}^2 B_{n-2}^2 + F_n^2 B_{n-1}^2 + B_n^2 \\ &\leq b_n V_1^2 + \sum_{i=1}^{n-1} B_i^2 b_{n-i} + B_n^2 \end{aligned} \tag{16}$$

where $$b_{n-i} = \prod_{k=i+1}^{n} F_k^2 \tag{17}$$

If we assume $V_1^2 < \infty$, it is readily seen that the right-hand side, and hence the left-hand side, of (16) becomes zero as $n \to \infty$. Thus,

$$\lim_{n\to\infty} E(X_n - \theta_n)^2 = 0 \tag{18}$$

The convergence of X_n to θ_n with probability one can be deduced by using the argument similar to Dvoretzky's proof for the stationary case (see end of section 3 and section 8 of [7]).

2.2 Mean Square Error and Optimal γ_n Sequence

A further use of (15) regarding the mean square error of the estimates is the following derivation of an optimal γ_n sequence. The result is a minimax solution since only the maximum possible mean square error will be minimized in carrying out the optimization process. From (15), we can write

$$\begin{aligned} V_{n+1}^2 &\leq F_n^2 V_n^2 + \gamma_n^2 \sigma^2 \\ &\leq \left(\frac{n+1}{n}\right)^2 (1 - \gamma_n)^2 V_n^2 + \gamma_n^2 \sigma^2 \end{aligned} \tag{19}$$

The minimum of the right-hand side of (19) is achieved at

$$\gamma_n = \frac{V_n^2}{V_n^2 + \sigma^2\left(\frac{n}{n+1}\right)^2} \tag{20}$$

Substituting the expression of (20) into (19) for γ_n, one obtains the following recursive relation for the mean square error

$$V_{n+1}^2 \leq \frac{\sigma^2 V_n^2}{V_n^2 + \sigma^2\left(\frac{n}{n+1}\right)^2} = \sigma^2 \gamma_n \tag{21}$$

Let the initial estimate have the mean square error $V_1^2 = E(X_1 - \theta_1)^2$ and define $\alpha = \sigma^2/V_1^2$. Using (20) and (21), we can iterate V_n^2 and γ_n alternately, obtaining the optimal sequence γ_n and minimized mean square error as follows

$$\gamma_n = \frac{(n+1)^2}{\frac{(n+1)(n+2)(2n+3)}{6} + (\alpha-1)} \tag{22}$$

$$V_{n+1}^2 = \sigma^2 \gamma_n = \frac{\sigma^2 (n+1)^2}{\frac{(n+1)(n+2)(2n+3)}{6} + (\alpha-1)} \tag{23}$$

It should be noted that the optimal γ_n sequence thus obtained can be easily shown to satisfy the conditions stated in (4) and a simple substitution of (22) for γ_n into (11) gives $\prod_{n=1}^{\infty} F_n = 0$.

Notice that the value of V_n^2 goes to zero, in the order of (1/n), as n increases without limit. The speed of convergence in terms of mean square error, however, is slightly slower than that of the convergence of estimates in the stationary case where only ordinary (one-step) stochastic approximation algorithm is employed, that is, (2) and (3) would be replaced by a single equation

$$X_{n+1} = X_n + \gamma_n (Y_{n+1} - X_n) \tag{24}$$

To clarify this argument, we shall make a quantitative comparison of these two algorithms by computing the mean square errors of the estimates resulting from both cases. Since the learning algorithm (24) has been previously treated [4], we simply display the corresponding expressions for optimal γ_n sequence and minimized mean square error V_n^2, respectively, as follows

$$\gamma_n = \frac{1}{n+\alpha} \tag{25}$$

$$V_{n+1}^2 = \frac{\sigma^2}{n+\alpha} \tag{26}$$

The result of these computations are shown in Table I. It is clear that the mean square error in the nonstationary case is greater except in the limiting case as n → ∞ where the error for both estimates vanishes.

TABLE I. COMPARISON OF MEAN SQUARE ERRORS ($\alpha = 1$)

No. of stage n	Stationary V_{n+1}^2	Nonstationary V_{n+1}^2
0 (initial est.)	σ^2	σ^2
1	$0.5\sigma^2$	$0.8\sigma^2$
2	$0.33\sigma^2$	$0.64\sigma^2$
3	$0.25\sigma^2$	$0.53\sigma^2$
4	$0.20\sigma^2$	$0.45\sigma^2$
5	$0.17\sigma^2$	$0.40\sigma^2$
6	$0.14\sigma^2$	$0.35\sigma^2$
⋮	⋮	⋮
10	$0.09\sigma^2$	$0.24\sigma^2$
20	$0.05\sigma^2$	$0.14\sigma^2$
30	$0.03\sigma^2$	$0.09\sigma^2$

Remarks. The discussion in this section has been restricted to the estimation of a time-varying mean of a probability distribution. The principle of the two-step stochastic approximation algorithm, however, is equally applicable to the learning of other parameters provided that they are also slowly varying in a linear or nearly linear fashion. Obvious modifications regarding the correction term in (3) must be made depending on the particular parameter under estimation.

III. NONSUPERVISED LEARNING IN NONSTATIONARY ENVIRONMENT

Nonsupervised learning deals with the problems where the learning samples are considered to be generated from a mixture distribution consisting of two or more component distributions. The learning samples thus generated are said to be unclassified, for the exact membership of component distribution to which a particular sample belongs is unknown. Our primary concern in this section is to estimate the unknown and time-varying parameters which characterize the mixture distribution as the learning samples are received. In particular, we shall investigate how the two-step stochastic approximation algorithm discussed above may be applied to such problems. The following examples are given to illustrate the various applications.

The problem statement:* Let $P(Y/m_{1,n};\ \sigma^2_{1,n})$ and $P(Y/m_{2,n};\ \sigma^2_{2,n})$ be two normal distributions characterized by their respective means $(m_{1,n};\ m_{2,n})$ and variances $(\sigma^2_{1,n};\ \sigma^2_{2,n})$. Let p_n and $1-p_n$ be the a priori probabilities for the two component distributions, respectively. We denote the time-varying parameter set $(m_{1,n};\ m_{2,n};\ \sigma^2_{1,n};\ \sigma^2_{2,n};\ p_n)$ by θ_n. The mixture distribution is then given by

$$P(Y/\theta_n) = p_n\ P(Y/m_{1,n};\ \sigma^2_{1,n}) + (1 - p_n)\ P(Y/m_{2,n};\ \sigma^2_{2,n}) \tag{27}$$

We wish to learn the unknown parameters through the unclassified learning samples $Y_1, Y_2, \ldots, Y_n$ generated in some fashion from $P(Y/\theta_n)$.

The partial solution: Let the first and second moment of Y with respect to $P(Y/\theta_n)$ be computed from (27) at the nth instant.

$$E_n(Y) = p_n m_{1,n} + (1 - p_n)\ m_{2,n} \tag{28}$$

$$E_n(Y^2) = p_n\ (m^2_{1,n} + \sigma^2_{1,n}) + (1 - p_n)(m^2_{2,n} + \sigma^2_{2,n}) \tag{29}$$

Example 1. Consider the simplest case where $m_{1,n}$ is the only time-varying parameter to be learned. Let $m_{2,n} = 0$; $p_n \doteq p$ and $\sigma^2_{1,n} = \sigma^2_{2,n} = \sigma^2$ are known. Let

$$m_{1,n+1} = (1 + n^{-1})\ m_{1,n} \tag{30}$$

which is a special case of (1) with μ arbitrarily large. From (28),

$$E_n(Y) = p\ m_{1,n} \tag{31}$$

Then we have

$$\begin{aligned} &E_{n+1}(Y) - (1 + n^{-1})\ E_n(Y) \\ &= p\ m_{1,n} - (1 + n^{-1})\ p\ m_{1,n} = p(m_{1,n} - (1 + n^{-1})\ m_{1,n}) \\ &= 0. \end{aligned} \tag{32}$$

That is, $E_n(Y)$ varies in a similar manner as $m_{1,n}$, hence satisfying the special case of condition (1) when μ is arbitrarily large. Now considering $E_n(Y)$ as a time-varying parameter of the mixture distribution and letting X_{n+1} be the (n+1)th estimate of $E_{n+1}(Y)$, we define

---

[*] The problem stated here is similar to that discussed in [4].

$$X_{n+1} = X_n^* + \gamma_n (Y_{n+1} - X_n^*) \tag{33}$$

where $X_n^* = (1 + n^{-1}) X_n$

Using the results established in the previous section, we obtain

$$\lim_{n\to\infty} E(X_n - E_n(Y)) = 0 \tag{34}$$

and $$P(\lim_{n\to\infty} X_n = E_n(Y)) = 1 \tag{35}$$

That is, $E_n(Y)$ and consequently $m_{1,n} = E_n(Y)/p$ from (31) is learned asymptotically in mean square and with probability one. It is also clear that this procedure can be applied to learn the unknown and time-varying p_n if $m_{1,n}$ becomes known and time-invariant.

Example 2. Learning the time-varying variance of the mixture distribution (27). Let $m_{2,n} = 0$; $m_{1,n} = m_1$ and $p_n = p$ are known. Let $\sigma_{1,n}^2 = \sigma_{2,n}^2 = \sigma_n^2$ be unknown and varied according to the following

$$\sigma_{n+1}^2 = (1 + n^{-1})\, \sigma_n^2 \tag{36}$$

Then (29) reduces to

$$E_{n+1}(Y^2) = p\, m_1^2 + \sigma_{n+1}^2 \tag{37}$$

Using (36) and (37) we have

$$E_{n+1}(Y^2) - (1 + n^{-1})\, E_n(Y^2)$$

$$= p\, m_1^2 + (1 + n^{-1})\, \sigma_n^2 - (1 + n^{-1})(p\, m_1^2 + \sigma_n^2)$$

$$= p\, m_1^2 - (1 + n^{-1})\, p\, m_1^2 = -\frac{\rho}{n} \tag{38}$$

where $\rho = p\, m_1^2$ = constant. Thus, $E_n(Y^2)$ again satisfies the condition in (1) with $\mu = 1$. In order to learn $E_n(Y^2)$ and hence σ_n^2 (see (37)), stochastic approximation algorithm similar to (33) in example 1 may be defined to obtain estimates which converge to the true parameter value in mean square and with probability one.

Remarks. (1) If, in addition to the unknown variance, one of the parameters p and m_1 is also unknown and possibly time-varying, then the constant $\rho = p\, m_1^2$ in the learning algorithm should be replaced by its successive estimates resulting from another learning algorithm such as the one discussed in example 1.

(2) The method given in the above examples can also be applied to learn unknown parameters of a mixture distribution with component distributions other than Normal. A discussion on the learning problems of this type may be found in [4].

IV. AN ACCELERATED DYNAMIC STOCHASTIC APPROXIMATION ALGORITHM

It was noted in section II that the convergence of estimates to the true parameter value for the dynamic algorithm would be slightly slower than that in the ordinary algorithm for the corresponding stationary case. This degradation of performance is clearly due to the presence of a time-varying trend in the parameter to be learned. Frequently, as a matter of practical significance, one would wish to speed up the convergence by taking advantage of the particular characteristic of the underlying parameter or of the distribution itself. As an illustrative example, let us consider a special situation of learning the mean of a distribution with μ arbitrarily large, i.e., $\theta_{n+1} + (1 + n^{-1})\ \theta_n$. Re-writing (3) we have

$$X_{n+1} - X_n^* = \gamma_n\ (Y_{n+1} - X_n^*) \tag{39}$$

It is seen that fewer fluctuations in the sign changes of $\gamma_n\ (Y_{n+1} - X_n^*) = (X_{n+1} - \theta_{n+1}) - (1 + n^{-1})(X_n - \theta_n)$ indicate that X_{n+1} is still far away from θ_{n+1}, whereas more frequent sign changes may indicate X_{n+1} is near θ_{n+1}. Consequently, using Kesten's acceleration scheme [8], we can modify the two-step algorithm in (2) and (3) to improve the speed of convergence. Define

$$X_{n+1} = X_n^* + \beta_n\ (Y_{n+1} - X_n^*) \tag{40}$$

where
$$X_n^* = (1 + n^{-1})\ X_n \tag{41}$$

$$\beta_1 = \gamma_1 \tag{42}$$

$$\beta_n = \gamma_{t(n)} \tag{43}$$

and
$$t(n) = 1 + \sum_{i=2}^{n} f\left[(X_{i+1} - X_i^*)(X_i - X_{i-1}^*)\right] \tag{44}$$

in which
$$f[z] = \begin{cases} 1 & \text{if} \quad z < 0 \\ 0 & \text{if} \quad z \geq 0 \end{cases} \tag{45}$$

In words, the modified algorithm takes another γ_n every time $(X_{i+1} - X_i^*)$ and $(X_i - X_{i-1}^*)$ differ in sign; otherwise, γ_n remains unchanged. The way that this modification is capable of speeding up the learning process can be readily seen from Table II. It is shown that when the sign changes are not so frequent (four times in the first ten stages), the total correction for the estimates of the modified algorithm reaches 3.7 units while the correction for the unmodified algorithm is only 2.8 units.

TABLE II. ILLUSTRATION OF POSSIBLE ACCELERATION

USING MODIFIED DYNAMIC ALGORITHM

$\alpha = 1, \quad \gamma_n = \dfrac{6(n + 1)^2}{(n + 1)(n + 2)(2n + 3)}$, all sign changes hypothetical.

(The same γ_n sequence obtained in sec. 1 is used for both algorithms.)

No. of stages n	1	2	3	4	5	6	7	8	9	10	Total Corrections
Sign of $(X_{n+1}-X_n^*)$	+	+	+	−	+	+	+	−	+	+	
Corrections (unmodified algorithm)	0.80	0.64	0.53	−0.45	0.40	0.35	0.31	−0.28	0.26	0.24	+2.80
Corrections (modified algorithm)	0.80	0.80	0.80	−0.64	0.53	0.53	0.53	−0.45	0.40	0.40	+3.70

V. DISCUSSIONS AND CONCLUSIONS

The two-step stochastic approximation algorithm discussed here has been shown to be capable of learning the time-varying parameters satisfying the condition $\theta_{n+1} - (1 + n^{-1})\,\theta_n = 0(n^{-\mu})$, $\mu \geq 1$. The primary requirement in using this algorithm is that one must have some knowledge about the functional form that the parameters supposedly vary with. For instance, we will have to assume that either $0(n^{-\mu})$ is known or it can be estimated from the learning samples. In the problems we have considered, the following cases are worth mentioning:

(1) Proportionality case: $\theta_n = \lambda n$, λ = const. and μ is arbitrarily large. Since $\theta_{n+1} - (1 + n^{-1})\,\theta_n = 0$, no prior knowledge is required except an initial guess of θ_1 with finite second moment.

(2) Linear case: $\theta_n = \lambda n + \rho$. We have $\theta_n - (1 + n^{-1})\,\theta_n = -\rho/n$ with $\mu = 1$. The prior knowledge required includes the true starting value of the time-varying parameter θ_n, i.e., $\rho = \theta_0$ when $n = 0$. It is then necessary during the learning process to estimate ρ by using another stochastic approximation algorithm, say, the Robbins-Monro procedure.

In general, let $\hat{0}_n\,(n^{-\mu})$ be the nth estimate of $0(n^{-\mu})$, then (2) and (3) should be revised to:

$$X_n^* = (1 + n^{-1})\,X_n + \hat{0}_n\,(n^{-\mu}) \qquad (2)'$$

$$X_{n+1} = X_n^* + \gamma_n\,(Y_{n+1} - X_n^*) \qquad (3)'$$

In this case, if ρ is unknown, it can, at least, be proved that the mean square error converges to a certain bound (non-zero bias) and the bound is a function of ρ. Further studies may be in the direction of seeking more general algorithms concerning time-varying functions other than linear or nearly linear.

REFERENCES

1. N. Abramson and D. Braverman, "Learning to recognize patterns in a random environment," IRE Trans. on Information Theory, vol. IT-8, pp. S-58 to S-63, September 1962.
2. D. G. Keehn, "A note on learning for gaussian properties," IEEE Trans. on Information Theory, vol. IT-11, pp. 126-132, January 1965.
3. V. Dupač, "A dynamic stochastic approximation method," Ann. Math. Stat., vol. 36, pp. 1695-1702, 1965.
4. Y. T. Chien and K. S. Fu, "On Bayesian learning and stochastic approximation," IEEE Trans. Systems Science and Cybernatics, vol. SSC-3, pp. 28-38, June 1967.
5. H. Robbins and S. Monro, "A stochastic approximation method," Ann. Math. Stat., vol. 22, pp. 400-407, 1951.
6. J. Kiefer and J. Wolfowitz, "Stochastic estimation of the maximum of a regression function," Ann. Math. Stat., vol. 23, pp. 462-466, 1952.
7. A. Dvoretzky, "On stochastic approximation," Proc. Third Berkeley Symp. on Math. Stat. and Prob., vol. 1, University of California Press, pp. 39-55, 1956
8. H. Kesten, "Accelerated stochastic approximation," Ann. Math. Stat., vol. 29, pp. 41-59, 1958.

ADAPTIVE STEP SIZE RANDOM SEARCH

M. A. SCHUMER AND K. STEIGLITZ
Electrical Engineering Department
Princeton University, Princeton, N.J.

ABSTRACT

Fixed step size random search for minimization of functions of several parameters is described and compared with the gradient method for a particular surface. A theoretical technique, using the optimum step size at each step, is analyzed. A practical adaptive step size random search algorithm is then proposed, and experimental experience is reported which shows the superiority of random search over other methods for sufficiently high dimension.

INTRODUCTION

The problem of locating the minimum of a function of several variables is one which arises frequently in many areas of technology, particularly in the design of adaptive control and communication systems.

The problem is the following: Given the quality function $Q(\underline{X})$, where $\underline{X}$ is a vector of adjustable parameters $x_1, \ldots, x_n$; find the value of $\underline{X}$ which minimizes Q. The following assumptions are made:

1. Q is unimodal. If it is not, a global search can be carried out first to partition the parameter space into regions where Q is unimodal.

2. The structure of the function $Q(\underline{X})$ is completely unknown. The only way that information can be obtained is by evaluating Q at specific points. This means, for example, that derivatives of Q are not directly measurable (if they exist at all).

3. The only significant cost involved in the operation of a search procedure results from evaluating Q. Therefore, the fewer function evaluations required, the more desirable is the procedure.

Of course any strategy suggested for solving the above problem can be evaluated only for a specific surface or class of surfaces. Analyses in this paper will be restricted to hyperspherical surfaces, and experimental results will be given for other surfaces as well.

Besides the many deterministic minimization algorithms developed, dating back to such classical methods as steepest descent, the Newton-Raphson method, and other gradient procedures; Brooks [1] and Rastrigin [2,3] have suggested randomized search strategies. Rastrigin has compared a fixed step size random search (FSSRS) method with a fixed step size gradient method and concluded that under certain circumstances FSSRS is superior. It is clear, however, that if the step size of the random search method were optimum at each step, even better performance would result. In this paper a hypothetical random search method, which uses the optimum step size at each point, will be analysed for a hyperspherical surface. An adaptive step size random search (ASSRS) method will then be proposed which approximates the performance of the optimum step size random search (OSSRS) procedure.

FIXED STEP SIZE RANDOM SEARCH (FSSRS)

The algorithm for FSSRS is

$$\underline{X}_{i+1} = \underline{X}_i - a_i \Delta\underline{X}_i + \Delta\underline{X}_{i+1} \tag{1}$$

where $\underline{X}_i$ is the position in state space at the i^{th} instant and $\Delta\underline{X}_i$ is a random vector of length s which is distributed uniformly over the hypersphere of radius s whose center is at the origin. The coefficient, a_i is given by

$$a_i = \begin{cases} 0 & \text{if } Q_i < Q^+_{i+1} \\ 1 & \text{if } Q_i \geq Q^+_{i+1} \end{cases} \tag{2}$$

where $Q_i = Q(\underline{X}_i)$ and $Q^+_{i-1} = \min\limits_{j=1,2,\ldots,i-1} Q_j$ is the smallest value of the quality function to be observed in the first i-1 steps. The coefficient a_i serves to negate the effect of an unsuccessful step. Rastrigin [2,3] has analyzed this algorithm for s=1 and the function $Q = [\sum\limits_{i=1}^{n} x_i^2]^{1/2}$. In order to motivate the development of a random search algorithm which adapts the step size, s, to the situation, we consider the performance of FSSRS as a function of s. Attention will be restricted to the function

$$Q(\underline{X}) = Q(\rho) = \sum_{i=1}^{n} x_i^2 = \rho^2 \tag{3}$$

which is smooth at its extremum and is more representative of problems with a minimum square error criterion than the function considered by Rastrigin. Otherwise, the analysis in this section is the same as Rastrigin.

Consider the plane formed by the displacement vector, $\Delta\underline{X}$, and the gradient vector through the starting point, A. [Figure 1] φ is the angle between the displacement vector and the negative gradient direction. φ_o is the largest value for φ for which there is an improvement as the result of the step, $\Delta\underline{X}$. For the assumed uniform distribution of displacement the probability density for φ considering φ only on $[0,\pi]$ (due to symmetry) is

$$p(\varphi) = \frac{\sin^{n-2}\varphi}{2\int_0^{\pi/2} \sin^{n-2}\varphi d\varphi} = \frac{\Gamma(n-1)}{2^{n-2}[\Gamma(\frac{n-1}{2})]^2} \sin^{n-2}\varphi \tag{4}$$

Search loss is defined as twice the ratio of the quality function to the expected value of the improvement per function evaluation. The search loss for the gradient technique is then

$$L_g(n,\eta) = \frac{2(n+1)}{2\eta-\eta^2} \tag{5}$$

where $\eta = s/\rho$, the ratio of step size to distance to the minimum. The search loss for FSSRS is found to be

$$L_r(n,\eta) = \frac{2\int_0^{\pi/2} \sin^{n-2} \varphi d\varphi}{\eta\int_0^{\varphi_0} \cos \varphi \sin^{n-2}\varphi d\varphi - \eta^2\int_0^{\varphi_0} \sin^{n-2}\varphi d\varphi} \qquad (6)$$

and φ_0 is equal to $\cos^{-1}(\eta/2)$.

Figure 2 shows the relative behavior of the random search and gradient methods for different values of η^{-1} and dimension, n. Above the boundary random search is superior (has a smaller search loss) to the gradient method, while below the boundary the gradient technique is superior. For n less than 4, the gradient technique is always superior, but for higher dimension random search is superior for small η.

OPTIMUM STEP SIZE RANDOM SEARCH (OSSRS)

If the step size for FSSRS is very small the probability of improvement is approximately 1/2, but the improvement is very small for a successful step, and this results in a small average improvement. On the other hand, if the step size is made too large the step will overshoot the minimum and the probability of improvement will be extremely small, also resulting in a very small average improvement. Somewhere between these extremes lies an optimum step size, i.e., a step size for which the expectation of the improvement of the quality function is maximum. At this step size the probability of success is not 1/2, but lies between 0 and 1/2.

The expected improvement, I, is equal to $2/L_r(n,\eta)$ and is given by

$$I(n,\eta) = \frac{\int_0^{\varphi_0} (2\eta \cos \varphi - \eta^2) \sin^{n-2}\varphi d\varphi}{2\int_0^{\pi/2} \sin^{n-2}\varphi d\varphi} \qquad (7)$$

In order to maximize I, the right side of eq. (7) is differentiated with respect to η and set equal to zero with the following equation for the optimal value of η resulting:

$$\int_0^{\varphi_0} \sin^{n-2}\varphi d\varphi = \frac{(1-\eta^2/4)^{\frac{n-1}{2}}}{n-1} \qquad (8)$$

Upon making the appropriate approximations, the following asymptotic expressions are found for large n

$$\eta_{opt} \sim 1.225/\sqrt{n} \qquad (9)$$

$$I_{max} \sim .406/n \qquad (10)$$

$$p_{opt} \sim .27 \qquad (11)$$

If the improvement, I, is proportional to 1/n (See eq. (10)), and if the relative improvements are independent as is true for OSSRS, then the average number of function evaluations for a fixed desired accuracy (relative to the starting value) is asymptotically linear in n, as is now demonstrated. Suppose that the search starts at a point at which the value of the quality function is Q_0, and that it is desired to terminate when the quality function reaches a final value of Q_f. As a result of the above assumptions, the expected value of the quality function, Q_M after M steps is

$$E[Q_M] = (1 - k/n)^M Q_0 \qquad (12)$$

where k is the constant of proportionality. Solving for the value of M for which $E[Q_M]$ is equal to the desired final value, Q_f results in

$$M = \frac{\log(Q_f/Q_0)}{\log(1-k/n)} \qquad (13)$$

Thus, the asymptotic expression for large n becomes

$$M \sim (\text{constant}) \cdot n \qquad (14)$$

where the constant is equal to $\frac{1}{k} \log(Q_f/Q_0)$.

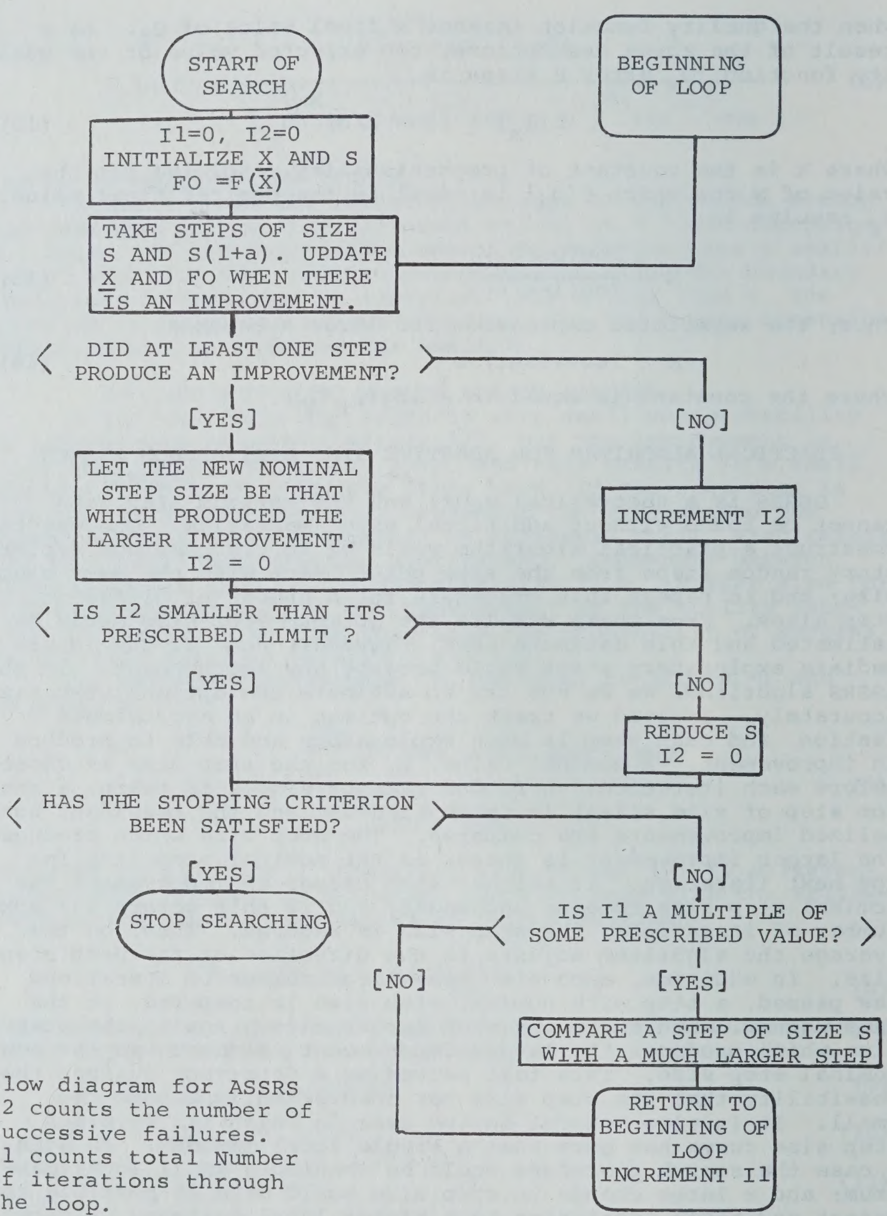
PRACTICAL ALGORITHM FOR ADAPTIVE STEP SIZE RANDOM SEARCH (ASSRS)

OSSRS is a theoretical model and the optimum step size cannot be found without additional experimentation. One way to construct a practical algorithm would be to try numerous exploratory random steps from the same point, each with the same step size; and to repeat this procedure for a number of different step sizes. From these results the optimum step size could be estimated and this estimate used. However, none of the intermediate exploratory steps would produce any improvement. In the ASSRS algorithm, we do not try to estimate the optimum step size accurately. Instead we track the optimum in an approximate fashion, and each step is both exploratory and able to produce an improvement. A nominal value, s, for the step size is chosen before each iteration. A random step of size s is taken, a random step of size s(1+a) is taken (1>a>0), and the resultant normalized improvements are compared. The step size which produces the larger improvement is chosen as the nominal step size for the next iteration. If neither step causes an improvement the nominal step size remains unchanged; and if this occurs for some number of iterations, the step size is reduced. Thus, on the average the algorithm adjusts to the direction of the best step size. In addition, each time some large number of iterations has passed, a step with nominal step size is compared, in the same manner, with a step of much larger size. Again, the step size which produces the larger improvement is chosen as the new nominal step size. This test serves as a deterrent against the possibility that the step size has inadvertantly become too small. It is also helpful in the case in which the I versus step size curve has more than a single local maximum. In such a case the search procedure could be chasing a small local maximum; and a large change in step size would make it possible to detect and begin adaptation to a higher local maximum. A flow diagram for the ASSRS algorithm is shown on the following page.

EXPERIMENTAL RESULTS

The ASSRS algorithm was tested on the IBM 7094 computer for a number of test functions. The results of these experiments are presented here.

The quality function $Q = \rho^2$ was tested to see how the ASSRS


START OF SEARCH
I1=0, I2=0
INITIALIZE X AND S
FO =F(X)
TAKE STEPS OF SIZE S AND S(1+a). UPDATE X AND FO WHEN THERE IS AN IMPROVEMENT.
BEGINNING OF LOOP
DID AT LEAST ONE STEP PRODUCE AN IMPROVEMENT?
[YES]
[NO]
LET THE NEW NOMINAL STEP SIZE BE THAT WHICH PRODUCED THE LARGER IMPROVEMENT. I2 = 0
INCREMENT I2
IS I2 SMALLER THAN ITS PRESCRIBED LIMIT ?
[YES]
[NO]
REDUCE S
I2 = 0
HAS THE STOPPING CRITERION BEEN SATISFIED?
[YES]
[NO]
STOP SEARCHING
IS I1 A MULTIPLE OF SOME PRESCRIBED VALUE?
[NO]
[YES]
COMPARE A STEP OF SIZE S WITH A MUCH LARGER STEP
RETURN TO BEGINNING OF LOOP INCREMENT I1

Flow diagram for ASSRS
I2 counts the number of successive failures.
I1 counts total Number of iterations through the loop.

algorithm compared to the Newton-Raphson method and to OSSRS. For each dimension from 1 to 40, 15 independent trials were run. The stopping criterion was $Q < 10^{-8}$ and the starting point was (1,1,...,1). The resulting average number of required function evaluations is well described by ICALL = 80n. Assuming that partial derivatives can be measured exactly by using finite differences, the Newton-Raphson method reaches the minimum after $(n+1)^2$ function evaluations. The results for ASSRS and for the Newton-Raphson method are shown in Figure 3. Extrapolating these curves, their intersection is at n = 78, beyond which dimension ASSRS is superior. If the fact that the partial derivatives cannot be measured exactly (which means that additional iterations are required) is taken into account, the Newton-Raphson curve becomes higher and the intersection with the ASSRS occurs at a smaller value of n. Also, if a lesser degree of accuracy is required, the ASSRS curve has a smaller slope while the Newton-Raphson curve is unchanged. This also results in a smaller value of n at the intersection of the curves, and ASSRS is then superior at a smaller value of dimension. As an indication of the variance in ICALL associated with ASSRS, the standard deviation for n = 100 was found to be 502 as compared with the mean of 7677.

On the basis of the experimental results, the average normalized improvement per step, I, was calculated for $Q = \rho^2$, and was found to be asymptotic to .2725/n. Thus, although the value of k is smaller than for OSSRS, the asymptotic form of I is still k/n; and the number of function calls was found to be asymptotically proportional to n as it is for OSSRS [See eq. (14)].

Figure 4 shows the results for $Q = \sum_{i=1}^{n} x_i^4$ averaged for 22 independent experiments for each dimension from 1 through 40. Again the starting point was (1,1,...,1). The stopping criterion was $Q < .5 \times 10^{-8}$. The Newton-Raphson method is again assumed to be able to measure derivatives without error, but the fact that Q is not quadratic is taken into account by multiplying the number of iterations required by $(n+1)^2$ in order to find the number of function calls. The results for the simplex method are those of Nelder and Mead [4], who found that for n=1 through 10, the number of function evaluations needed for the simplex method is well described by $3.16(n+1)^{2.11}$. This formula was extrapolated to higher dimensions and plotted along with the curves for ASSRS and Newton-Raphson. In this case, ASSRS is superior to Newton-Raphson for n > 2 and to the simplex method for n > 10. For ASSRS, ICALL is again proportional to n and I is asymptotic to .427/n.

The quadratic form, $Q = \sum_{i=1}^{n} a_i x_i^2$, was also used as a test function, with the coefficients $a_1, a_2, \ldots, a_n$ randomly chosen from a uniform distribution on [.1, 1]. The minimization was repeated twenty times for each dimension from 1 to 20 and six times for n = 100. The search was terminated when Q was less than one-thousandth of its initial value. Figure 5 shows the resulting average number of function evaluations required as a function of dimension. The result for 100 dimensions was ICALL = 3396. Again, ICALL was found to be approximately linear in n.

Rastrigin [3] has done some work with random search algorithms which adapt only to the best direction with a fixed step size. The problem with this method is that convergence is only guaranteed to within a distance of one step size to the minimum, so that if a high accuracy is desired the steps must be small, thus requiring many function evaluations. Adapting the step size seems to be more fruitful than adapting to the direction. A combination of the two methods, however, would seem to be in order and should be the subject of future investigations.

ASSRS was also tested using Rosenbrock's function, $Q = 100(x_2-x_1{}^2)^2 + (1-x_1)^2$ and although it converged ASSRS was inferior to Rosenbrock's method [5] and Powell's method [6] for this function. It should, therefore, be noted that ASSRS is not very effective as a ridge follower, but shows its superiority in multi-dimensional problems without narrow valleys or ridges. Combining directional adaptation with step size adaptation may result in removing this limitation.

CONCLUSION

In this presentation, the problem of minimizing a function of several parameters by the method of random search has been discussed. The fixed step size random search algorithm has been described and compared to the simple gradient technique on the basis of search loss. It has been shown that for $n > 4$, FSSRS is superior for sufficiently small η. Optimum step size random search was introduced and its performance investigated for hyperspherical surfaces.

A practical algorithm for adaptive step size random search has been described and compared with deterministic methods. For the functions $Q = \sum_{i=1}^{n} x_i^2$, $Q = \sum_{i=1}^{n} x_i^4$ and $Q = \sum_{i=1}^{n} a_i x_i^2$ the number of function evaluations required for a desired accuracy for the deterministic methods increases at a rate that is proportional to at least the second power of n; and the computation time increases as n^3. (This is true for other classical methods besides the Newton-Raphson and simplex methods. See reference [7]). However, for ASSRS the number of required function evaluations is proportional to n, and the computation time is proportional to n^2. The computation time could be made proportional to n if parallel computations were used. Thus, the conclusion is reached, that despite its simplicity, adaptive random search is an attractive technique for problems with large numbers of dimensions.

ACKNOWLEDGMENT

This work was partially supported by Army Research Office-Durham under contract no. DA-31-124-ARO-D-292, by a National Science Foundation Graduate Fellowship, and by a research grant from the Bendix Corporation. It made use of computer facilities supported in part by National Science Foundation Grant NSF-GP-579.

REFERENCES

1. S. H. Brooks, "Discussion of Random Methods for Locating Surface Maxima", Operations Research, vol. 6, pp. 244-251; 1958.

2. L. A. Rastrigin, "The Convergence of the Random Search Method in the Extremal Control of a Many Parameter System",

Automation and Remote Control, vol. 24, no. 11, pp. 1337-1342; November, 1963.

3. L.A. Rastrigin, *Sluchainyi Poisk v Zadachakh Optimizatsii Mnogoparametricheskikh Sistem*, Akademiia Nauk Latviiskoi SSR; Riga, USSR, 1965.

4. J. A. Nelder and R. Mead, "A Simplex Method for Function Minimization", *The Computer Journal*, vol. 7, no. 4, pp. 308-312; January, 1965.

5. H. H. Rosenbrock, "An Automatic Method for Finding the Greatest or Least Value of a Function", *The Computer Journal*, vol. 3, no. 3, pp. 175-184; October, 1960.

6. M. J. D. Powell, "An Iterative Method for Finding the Stationary Values of a Function of Several Variables", *The Computer Journal*, vol. 5, no. 2, pp. 147-151; July, 1962.

7. D. J. Wilde, *Optimum Seeking Methods*, Prentice-Hall; Englewood Cliffs, N.J.; 1964.

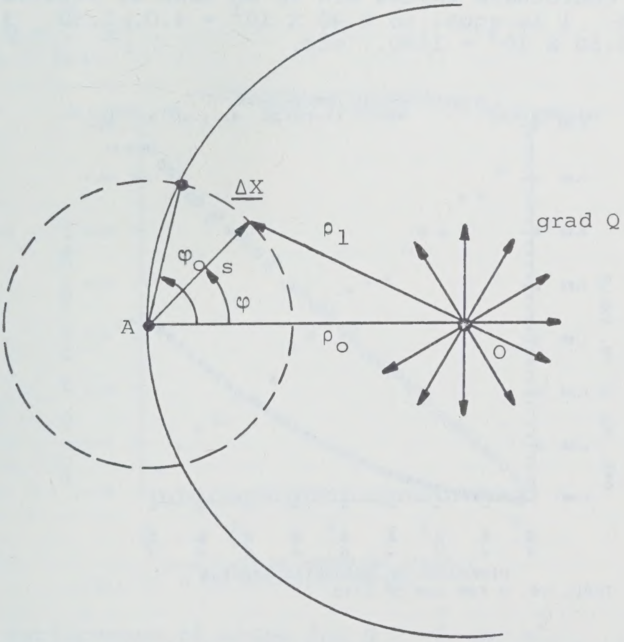


Figure 1. A portion of parameter space illustrating the method of random search.

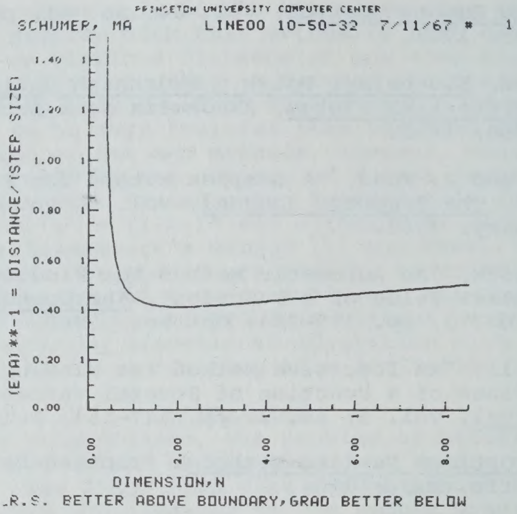


Figure 2. The performance boundary between the FSSRS and gradient methods.

NOTE: The coordinate values are to be read as follows: 0.40 1 is equal to 0.40 X 10^1 = 4.0, 1.50 3 is equal to 1.50 X 10^3 = 1500., etc.

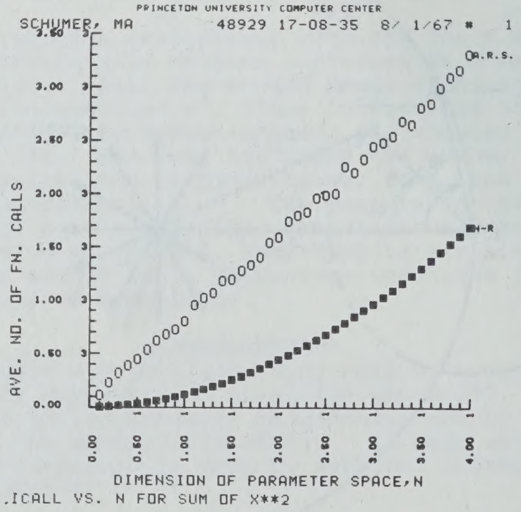


Figure 3. Performance of ASSRS (0's) and the Newton-Raphson method (squares) for

$$Q = \sum_{i=1}^{n} x_i^2 .$$

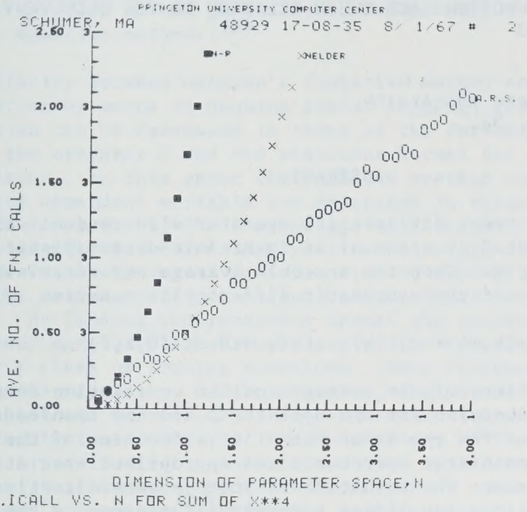


Figure 4. Performance of ASSRS (0's), the Newton-Raphson method (squares) and the Simplex method (X's) for $Q = \sum_{i=1}^{n} x_i^4$.

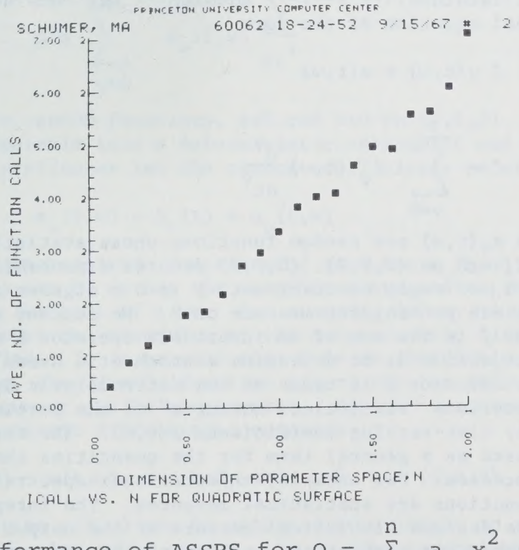


Figure 5. Performance of ASSRS for $Q = \sum_{i=1}^{n} a_i x_i^2$.

STOCHASTIC GREEN'S FUNCTIONS AND THEIR RELATION TO THE RESOLVENT KERNELS OF INTEGRAL EQUATIONS

LEON H. SIBUL
Pennsylvania State University
University Park, Pa.

ABSTRACT

Let $\mathcal{L}$ be an n order differential operator with randomly time varying coefficients such that $\mathcal{L}$ is a sum of an invertible deterministic operator L and a random operator R. Then the ensemble average and correlation functions of the solution of the stochastic differential equation

$$\mathcal{L}y(t,\omega) = x(t,\omega) \quad t\epsilon T,\ \omega\epsilon\Omega \text{ on } (\Omega,F,P)$$

can be expressed in terms of the average and the correlation functions of $x(t,\omega)$, the Green's function for the operator L and the resolvent kernel of an integral equation. The resolvent kernel is a function of the Green's function of the deterministic operator L and appropriate statistics of the stochastic coefficients. The results constitute a generalization of the corresponding expressions for linear time-varying systems to the linear randomly time-varying systems.

INTRODUCTION

Differential equations with randomly time-varying coefficients arise naturally in many practical problems. Many examples of these have been discussed in the literature.(1,4,13,14) Adomian(3) has considered a stochastic differential equation of the form

$$\mathcal{L}\, y(t,\omega) = x(t,\omega)$$

where

$$\mathcal{L} = \sum_{\nu=0}^{n} a_\nu\,(t,\omega)\,\frac{d^\nu}{dt^\nu}$$

and $x(t,\omega)$ and the $a_\nu(t,\omega)$ are random functions whose statistics are known and defined on $t\epsilon T$, $\omega\epsilon\Omega$ on (Ω,F,P). (Ω,F,P) denotes a probability measure space, i.e., Ω is a non-empty abstract set, F is a σ algebra of subsets of Ω, and P is a complete probability measure on F. He further assumes that the random operator $\mathcal{L}$ is the sum of an invertible operator L and a random operator R. His objective is to determine a stochastic Green's function(2) for the stochastic operator $\mathcal{L}$ in terms of the deterministic Green's function for L and the appropriate "statistical measures" of the forcing function $x(t,\omega)$ and randomly time-varying coefficients $a(t,\omega)$. The term "statistical measures" is used as a general term for the quantities that characterize stochastic processes. For example, expectations, spectral densities and correlation functions are statistical measures. The integral kernel which expresses the desired statistical measure of the output process in terms of the corresponding statistical measure of the input and appropriate statistical measures of the stochastic coefficients is called the "stochastic Green's function". Adomian develops an iterative process for finding the stochastic Green's functions for expectation and the auto-correlation function of the output process. This iterative method achieves the desired separations of ensemble averages without restrictions to perturbation-type approaches or recourse to the generally non-valid closure approximations of the hierarchy equation methods.(3)

The similarity between Adomian's iterative method and solution of an integral equation by means of Neumann series suggests that the stochastic Green's function can be expressed in terms of the deterministic Green's function for the operator L and the resolvent kernel for the Volterra integral equation. In this paper the ensemble average and correlation function of the dependent variable are expressed in terms of the Green's function of the operator L, the average and correlation function of the forcing function $x(t,\omega)$ and the resolvent kernel of the Volterra integral equation. The resolvent kernel is a function of appropriate statistical measures of the random coefficients and Green's function of the deterministic operator. By finding the resolvent kernel the solution of the problem of finding means and correlation functions of the dependent variable is expressed for a class of forcing functions. Many different methods such as Neumann series expansion, Fredholm theory, Schmidt-Hilbert theory or numerical methods can be used to find the resolvent kernel.(7,10,12) Even though the practical usefulness of the results may be limited by the large amount of labor required for calculation of the resolvent kernels, the results are still of interest because these results connect the theory of stochastic Green's functions(2) with the well-developed theory of Fredholm integral equations.(7,10,12)

STOCHASTIC GREEN'S FUNCTION AND RESOLVENT KERNEL

Following Adomian's(3) approach, let $\mathfrak{L}$ be an n order stochastic differential operator such that

$$\mathfrak{L} \equiv \sum_{\nu=0}^{n} a_\nu(t,\omega) \frac{d^\nu}{dt^\nu} \tag{1}$$

where $a_\nu(t,\omega)$ are random functions, $t \epsilon T$ and $\omega \epsilon \Omega$ on (Ω,F,P). Let the operator $\mathfrak{L}$ be separable into a deterministic operator L and a random operator R. In particular let the random coefficients be of the form

$$a_\nu(t,\omega) = \beta_\nu(t) + \alpha_\nu(t,\omega) \tag{1a}$$

where $\beta_\nu(t)$ are deterministic functions of time and $a_\nu(t,\omega)$ are stochastic processes. $\beta_\nu(t)$ can be either the ensemble average of $a_\nu(t,\omega)$ or some other convenient function of time. For example, it may be possible to choose $\beta_\nu(t)$ so that the inversion of the deterministic differential operator L is simplified. It is assumed that L is an invertible differential operator; that is, the Green's function $G(t,\tau)$ for the differential operator L is known or can be constructed. One can associate with the operator $\mathfrak{L}$ a stochastic differential equation

$$\mathfrak{L}\, y(t,\omega) = x(t,\omega) \qquad t \epsilon T \quad \text{and} \quad \omega \epsilon \Omega \quad \text{on} \quad (\Omega,F,P) \quad . \tag{2}$$

The forcing function $x(t,\omega)$ can be either deterministic or random. For greater generality, let it be random. It is also assumed that $x(t,\omega)$ is statistically independent of the random coefficients. By the assumption that $\mathfrak{L} = L + R$ the equation (2) can be written as follows:

$$L\, y(t,\omega) = x(t,\omega) - R\, y(t,\omega) \tag{3}$$

where,

$$L = \sum_{\nu=0}^{n} \beta_{\nu}(t) \frac{d^{\nu}}{dt^{\nu}} \tag{3a}$$

and

$$R = \sum_{\nu=0}^{n} \alpha_{\nu}(t,\omega) \frac{d^{\nu}}{dt^{\nu}} \tag{3b}$$

By the assumption that the Green's function for the deterministic operator L is known, (3) can be converted into an integral equation:

$$y(t,\omega) = L^{-1} x(t,\omega) - L^{-1} R\, y(t,\omega) \tag{4}$$

where,

$$L^{-1} = \int_{o}^{t} d\tau\, G(t,\tau) \tag{4a}$$

and

$$L^{-1}R = \int_{o}^{t} d\tau\, G(t,\tau) \sum_{\nu=0}^{n} \alpha_{\nu}(\tau,\omega) \frac{d^{\nu}}{d\tau^{\nu}} \quad . \tag{4b}$$

The use of 0 as the lower limit assumes that either the system was initially at rest or the initial conditions have been taken into account in the construction of the Green's function $G(t,\tau)$. The upper limit is t for causal systems, since for causal systems $G(t,\tau) = 0$ for $t < \tau$. To simplify the notation, let

$$L^{-1} x(t,\omega) = F(t,\omega) \tag{5a}$$

and let

$$K(t,\tau) = G(t,\tau) \sum_{\nu=0}^{n} a_{\nu}(\tau,\omega) \frac{d^{\nu}}{d\tau^{\nu}} \quad . \tag{5b}$$

With this notation equation (4) can be expressed as an integral equation which resembles the Volterra integral equation

$$y(t,\omega) = F(t,\omega) + \lambda \int_{o}^{t} K(t,\tau)\, y(\tau,\omega)\, d\tau . \tag{6}$$

In this case $\lambda = -1$. The solution to equation (6) can be expressed analogously to the Neumann series solution in terms of iterated kernels:

$$y(t,\omega) = F(t,\omega) + \sum_{m=1}^{\infty} \lambda^{m} \int_{o}^{t} K_{m}(t,\tau)\, F(\tau,\omega)\, d\tau \tag{7}$$

where $K_m(t,\tau)$ is defined by the recurrence formula

$$K_m(t,\tau) = \int_o^t K(t,\tau_1) K_{m-1}(\tau_1,\tau) \, d\tau_1 \tag{8}$$

and

$$K_1(t,\tau) = K(t,\tau) \, . \tag{8a}$$

If the order of summation and integration can be interchanged in (7) it becomes

$$y(t,\omega) = F(t,\omega) + \lambda \int_o^t \left[\sum_{m=0}^{\infty} \lambda^m K_{m+1}(t,\tau) \right] F(\tau,\omega) \, d\tau \tag{9}$$

Then equation (9) can be written in terms of the resolvent kernel $\Gamma(t,\tau,\lambda,\omega)$ as follows:

$$y(t,\omega) = F(t,\omega) + \lambda \int_o^t \Gamma(t,\tau,\lambda,\omega) \, F(\tau,\omega) \, d\tau \tag{10}$$

where

$$\Gamma(t,\tau,\lambda,\omega) = \sum_{m=0}^{\infty} \lambda^m \; K_{m+1} \; (t,\tau,\lambda,\omega) \; . \tag{10a}$$

If the order of integration and summation cannot be interchanged in (7) other methods, such as Fredholm theory, can be used for construction of the resolvent kernel. Since in this case λ is -1, the resolvent kernel is written from now on without λ in its argument. At this point we have nothing essentially different from classical differential equation theory. The difficulty arises since $y(t,\omega)$ is not a physically significant quantity, only its statistical measures are. The mean value of the dependent variable $y(t,\omega)$ can be computed by taking the ensemble average of equation (10). It must be noted that because of the assumption that the random coefficients and forcing function are independent the problem simplifies in that the resolvent kernel $\Gamma(t,\tau,\omega)$ and $F(t,\omega)$ are statistically independent of one another. Hence, the ensemble averages separate and the ensemble average of $y(t,\omega)$ is given by:

$$<y(t,\omega)> = < F(t,\omega)> - \int_o^t < \Gamma \; (t,\tau,\omega) > < F(\tau,\omega) > d\tau \tag{11}$$

where

$$< F(t,\omega) > = \int_o^t G(t,\tau) < x(t,\omega) > d\tau \qquad . \tag{11a}$$

In this case the stochastic Green's function is simply $< \Gamma(t,\tau,\omega)>$.

Similarly, the autocorrelation function of $y(t,\omega)$ can be computed. For greater generality $y(t,\omega)$ is taken to be a complex function. Then, the auto-correlation function of $y(t,\omega)$ denoted by $R_{yy}(t_1 t_2)$ is:

$$R_{yy}(t_1 t_2) = \langle y(t_1,\omega)\, y^*(t_2,\omega) \rangle$$

$$= \langle (F(t_1,\omega) - \int_o^{t_1} \Gamma(t_1,\tau_1,\omega)\, F(\tau_1,\omega)\, d\tau_1)$$

$$(F^*(t_2,\omega) - \int_o^{t_2} \Gamma^*(t_2,\tau_2,\omega)\, F^*(\tau_2,\omega)\, d\tau_2) \rangle$$

$$= \langle F(t_1,\omega)\, F^*(t_2,\omega) \rangle$$

$$- \int_o^{t_1} \langle \Gamma(t_1,\tau_1,\omega) \rangle \langle F(\tau_1,\omega)\, F^*(t_2,\omega) \rangle\, d\tau_1$$

$$- \int_o^{t_2} \langle \Gamma^*(t_2,\tau_2,\omega) \rangle \langle F(t_1,\omega)\, F^*(\tau_2,\omega) \rangle\, d\tau_2$$

$$- \int_o^{t_1} \int_o^{t_2} \langle \Gamma(t_1,\tau_1,\omega)\, \Gamma^*(t_2,\tau_2,\omega) \rangle \langle F(\tau_1,\omega)\, F^*(\tau_2,\omega) \rangle\, d\tau_1\, d\tau_2 \qquad (12)$$

The quantity $\langle F(t_1,\omega)\, F^*(t_2,\omega) \rangle$ can be denoted by $R_{FF}(t_1,t_2)$ and (12) can be written more compactly by

$$R_{yy}(t_1,t_2) = R_{FF}(t_1,t_2)$$

$$- \int_o^{t_1} \langle \Gamma(t_1,\tau_1,\omega) \rangle\, R_{FF}(\tau_1,t_2)\, d\tau_1$$

$$- \int_o^{t_2} \langle \Gamma^*(t_2,\tau_2,\omega) \rangle\, R_{FF}(t_1,\tau_2)\, d\tau_2$$

$$+ \int_o^{t_1} \int_o^{t_2} \langle \Gamma(t_1,\tau_1,\omega)\, \Gamma^*(t_2,\tau_2,\omega) \rangle R_{FF}(\tau_1,\tau_2)\, d\tau_1, d\tau_2 \qquad (12a)$$

where

$$R_{FF}(t_1,t_2) = \int_o^{t_1}\int_o^{t_2} G(t_1,\tau_1)G^*(t_2,\tau_2) < x(\tau_1,\omega)x^*(\tau_2,\omega) > d\tau_1\, d\tau_2$$

$$= \int_o^{t_1}\int_o^{t_2} G(t_1,\tau_1)G^*(t_2,\tau_2)\, R_{xx}(\tau_1,\tau_2)\, d\tau_1\, d\tau_2 \quad . \tag{12b}$$

Equation (12) is very messy looking, yet it gives one of the most important properties of y. It is unfortunate that many of the interesting properties of y are given by non-linear functions. If the coefficients of the differential operator are strictly deterministic, then the last three terms in (12a) vanish and $R_{yy}(t_1,t_2)$ is given by (12b), which agrees with the well known result for non-random systems. The cross correlation between the output and input can be also found by using equation (10)

$$R_{yx}(t_1,t_2) = < y(t_1,\omega)\, x^*(t_2,\omega) >$$

$$= \int_o^{t_1} G(t_1,\tau_1)\, R_{xx}(\tau_1,t_2)\, d\tau_1$$

$$- \int_o^{t_1} < \Gamma(t_1,\tau_1,\omega) > \left(\int_o^{\tau_1} G(\tau_1,\tau_o)\, R_{xx}(\tau_o,t_2)\, d\tau_o \right) d\tau_1 \tag{13}$$

Examination of equations (11), (12a) and (13) sheds some light on the stochastic Green's functions$^{(2)}$ for a differential operator which can be separated into a sum of deterministic operator L and random operator R. If it is desired, the stochastic Green's functions can be identified in (11), (12a) and (13). In all three cases, the first term in the expressions is due to the deterministic operator alone. If the randomness of the coefficients of $\mathcal{L}$ vanishes, only the first terms remain. The succeeding terms are complicated functions of random coefficients and deterministic Green's function for the operator L. When higher order coefficients ($\nu \geq 1$) are random, construction of the resolvent kernel involves differentiation as indicated by (5b). This is not too surprising for a relatively complicated problem. In other physical problems similar complications arise. For example, an expression for the dyadic Green's function associated with the solution of a vector wave equation involves differentiation.$^{(11)}$

In the actual solution of problems, construction of the resolvent kernel presents major difficulties. As shown by Adomian$^{(3)}$, Neumann series type if iteration can be successfully used. This type of iteration has the advantage that the previously computed term is used in the next term and so on. One can stop the iteration at any time when the remainder term becomes smaller than a prescribed value. By suitably redefining the kernel $K(t,\tau)$ the Volterra integral equation can be considered as a special form of the Fredholm integral equation. There exists a well developed theory of non-random Fredholm integral equations.$^{(7,10,12)}$ Bharucha-Reid$^{(5,6)}$ has considered the solution of the following random Fredholm integral equation

$$\int_{D_\omega} K(t,u)\ y(u,\omega)\ dm(u) - \lambda(u,\omega) = x(t,\omega) \qquad (14)$$

He uses the theory of random contraction transformations of O. Hanš(8,9) and A. Špaček(15) to prove the existence of the solution of (14). The equation (6) is a more complicated case, because here the kernel of the integral equation is also a random function.

When only the lowest order coefficient a_o is a random function, the construction of the resolvent kernel is greatly simplified. Iterations proceed in a straightforward manner as shown by Adomian(3). This method is probably the easiest and most straightforward method of solution. Other methods can also be used to compute the resolvent kernel. For example, in the case of a Hermetian kernel, the Hilbert-Schmidt formula can be used to compute the resolvent kernel. It is possible to use Fredholm theory for expressing the resolvent kernel as a ratio of two infinite power series, both of which converge, even when the Neumann series expansion does not converge. From the practical calculation standpoint, Fredholm theory usually involves a prohibitive amount of labor and hence simpler methods are preferable. Nevertheless, Fredholm theory has provided valuable information concerning existence and uniqueness of solutions of Fredholm integral equations. Development of the resolvent kernel by Fredholm series is strongly suggestive of matrix inversion and further consideration of this approach may lead to the insights which are useful for finding stochastic Green's functions by the matrix method.(1,2) Numerical methods may also be used to solve the equation (6) or to obtain an approximate expression for the resolvent kernel.

CONCLUSIONS

The expressions (11), (12) and (13) constitute a generalization of the corresponding expressions for the linear time-varying systems to the linear randomly time varying systems. Even though computation of the resolvent kernel is a major problem, the study of the resolvent kernel of integral equation (6) seems to be fruitful in understanding the general nature of stochastic Green's functions. Results presented here may be generalized to the partial differential equations and may be used to study wave propagation in randomly time and space varying media.

ACKNOWLEDGMENTS

I am grateful to Professors C. Comstock and G. McMurtry for reading and discussing this paper.

REFERENCES

1. G. Adomian, "Linear Stochastic Operators", Ph.D. (Physics) Dissertation, University of California (Los Angeles) 1961

2. G. Adomian, "Stochastic Green's Functions", Annual Symposia in Applied Math., Vol. XVI Amer. Math. Soc., 1964

3. G. Adomian, "Theory of Random Systems", Trans. 4th Prague Conf. Information Theory, pp. 205-222. House Czechoslovak Acad. Sci., Prague, 1967

4. K. J. Åström, "On a First-Order Stochastic Differential Equation" Int. Journal of Control, Vol. 1, No. 4, pp. 301-326, April 1965

5. A. T. Bharucha-Reid, "On Random Solution of Fredholm Integral Equations", Bull. Amer. Math. Soc. 66, (1960), 104-109

6. A. T. Bharucha-Reid, "On Random Solutions of Integral Equations in Banach Spaces". Trans. 2nd Prague Conf. Information Theory, pp. 27-28, Publ. House Czechoslovak Acad. Sci. Prague 1960

7. R. Courant and D. Hilbert, "Methods of Mathematical Physics", Vol. I, Interscience, New York 1953

8. O. Hanš, "Reduzierende Zufällige Transformation", Czechoslovak Math. J. Vol. 7, (1957) pp. 154-158

9. O. Hanš, "Random Fixed Point Theorems" Trans. 1st Prague Conf. Information Theory, pp. 105-125, Publ. House Czechoslavak Acad. Sci., Prague, 1957

10. F. B. Hildebrand, "Methods of Applied Mathematics", Prentice-Hall, Inc., New York 1952

11. H. Levine and J. Schwinger, "On the Theory of Electro-magnetic Wave Defraction by an Aperture in an Infinite Plane Conducting Screen", in "Theory of Electro-magnetic Waves", Symposium, Interscience New York 1951

12. W. V. Lovitt, "Linear Integral Equations", McGraw-Hill Book Co., Inc., New York 1924

13. A. Rosenblom, T. Heilfron, and D. L. Trautman, "Analysis of Linear Systems with Randomly Varying Inputs and Parameters", IRE Convention Record 4, 106 1954

14. J. C. Samuels and A. C. Eringen, "On Stochastic Linear Systems", J. Math and Phys. 38 1959

15. A. Špaček, "Zufällige Gleichungen", Czechoslovak Math. J. Vol. 5 1955 pp. 462-466

AN UPPER BOUND ON THE ZERO-CROSSING DISTRIBUTION

NICHOLAS A. STRAKHOV and LUDWIK KURZ
Bell Telephone Laboratories
Murray Hill, New Jersey 07971
New York University
Bronx, New York 10453

ABSTRACT

Let Q(T) equal the probability that a random process, x(t), does not cross the zero axis for a randomly selected interval of length T. A family of upper bounds for Q(T) is derived with only mild restrictions imposed on x(t) and it is shown that for Gaussian random processes only one member of the family provides useful formulae. Specific results are obtained for x(t) representing a number of interesting random processes.

INTRODUCTION

Let Q(T) equal the probability that a random process, x(t), does not cross the zero axis for a randomly selected interval of length T. The problem of determining Q(T) (and related functions) has important applications in communications theory and has been investigated by many authors, [1], [4]-[7], [9] to name a few. Reference [7] gives an extensive bibliography of most of the related work on this subject prior to 1958. Despite all this effort, Q(T) is known only when x(t) is a simple non-Gaussian process (such as a process whose zero-crossings obey the Poisson distribution) or a stationary Gaussian zero-mean process with one of four explicit correlation functions [7], [9]. Most of the rest of the results obtained are either approximate or form upper or lower bounds [7].

In this paper, a whole family of upper bounds on Q(T) is developed. For computational purposes, however, only one member of the family has been found to provide useful results for most cases of interest.

DERIVATION OF AN UPPER BOUND ON Q(T)

Consider the transformation

$$z_T = \frac{1}{T} \int_0^T \operatorname{sgn}[x(t)]dt \tag{1}$$

where x(t) is a sample function of a stochastic process

T is a fixed observation interval

and

z_T is a random variable defined by the stochastic integral (1).

The function sgn[x(t)] is defined as

$$\operatorname{sgn}[x] = \begin{cases} +1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases}$$

Since z_T is a continuous random variable[*] (except at $z_T = \pm 1$), it has a probability density function, $p(z_T)$, associated with it. From (1), two properties of $p(z_T)$ are immediately apparent, regardless of the statistics governing $x(t)$

a)
$$p(z_T) = 0 \quad \text{for} \quad |z_T| > 1 \tag{2}$$

b)
$$\lim_{\varepsilon\to 0} [P(T+\varepsilon) - P(T-\varepsilon)] = Q_U(T) \tag{3}$$

and

$$\lim_{\varepsilon\to 0} [P(-T+\varepsilon) - P(-T-\varepsilon)] = Q_L(T) \tag{4}$$

where

$$P(\alpha) = \int_{-\infty}^{\alpha} p(z_T)dz_T ,$$

$$Q_U(T) = \text{Prob}\{x(t) \geq 0 \text{ for } 0 \leq t \leq T\}, \tag{5}$$

and

$$Q_L(T) = \text{Prob}\{x(t) \leq 0 \text{ for } 0 \leq t \leq T\} \tag{6}$$

Obviously, $Q(T)$ as defined previously is related to the last two quantities by

$$Q(T) = Q_U(T) + Q_L(T) \tag{7}$$

If $x(t)$ is a symmetric[†] process, then

$$Q_U(T) = Q_L(T) = \tfrac{1}{2}Q(T) \tag{8}$$

As a consequence of properties (a) and (b), $p(z_T)$ can be represented by

$$p(z_T) = g(z_T) + Q_L(T)\delta(z_T+1) + Q_U(T)\delta(z_T-1) \tag{9}$$

where $g(z_T) = 0$ for $|z_T| > 1$

and $\delta(x)$ is the Dirac delta function.

Next, consider the even order moments of $p(z_T)$, denoted by

$$q_{2k} = \int_{-1}^{1} z_T^{2k} p(z_T)dz_T \tag{10}$$

$$(k = 0,1,2,\ldots)$$

---

[*] In this paper, pathological cases where z_T is not a continuous random variable are ignored.

[†] The stocastic process $x(t)$ will be called symmetric if the probability measures that govern it also govern the process, $-x(t)$.

By substituting (9) into (10) one obtains

$$q_{2k} = \int_{-1}^{1} z_T^{2k} g(z_T) dz_T + Q_U(T) + Q_L(T) \tag{11}$$

$$k = 0,1,2,\ldots$$

Neglecting the first term in the right-hand side of (11) (which is always positive) and taking (7) into account leads to a family of upper bounds for Q(T) expressed by

$$Q(T) \leq q_{2k} \tag{12}$$

$$k = 0,1,2,\ldots$$

For k = 0, (12) reduces to the obvious result

$$Q(T) \leq 1$$

Before discussing the usefulness of the inequality (12), an expression for the moments will be derived.

From its definition, (10), q_{2k} can be expressed as

$$q_{2k} = E\left\{z_T^{2k}\right\}$$

where $E\{\cdot\}$ denotes the expected value of the quantity enclosed in braces. Substitution for z_T from (1) results in

$$q_{2k} = \frac{1}{T^{2k}} E\left\{\int_0^T \int_0^T \ldots \int_0^T y(t_1)y(t_2)\ldots y(t_{2k})dt_1 dt_2 \ldots dt_{2k}\right\} \tag{13}$$

where

$$y(t_i) = \mathrm{sgn}[x(t_i)], \quad i = 1,2,\ldots,2k$$

Interchanging the order of integration and expectation yields

$$q_{2k} = \frac{1}{T^{2k}} \int_0^T \int_0^T \ldots \int_0^T R(t_1,t_2,\ldots,t_{2k})dt_1 dt_2 \ldots dt_{2k} \tag{14}$$

where

$$R(t_1,t_2,\ldots,t_{2k}) = E\{y(t_1)y(t_2)\ldots y(t_{2k})\} \tag{15}$$

which completes the derivation of (12).

We now make some remarks concerning the ordering of the family of inequalities expressed by (12).

Denote the first term in the right-hand side of (11) by s_{2k}, or

$$s_{2k} = \int_{-T}^{T} z_T^{2k} g(z_T) dz_T \tag{16}$$

We next establish that $s_{2k} > s_{2k+2}$ $(k = 0,1,2,\ldots)$ which in turn establishes

$$1 > q_2 > q_4 \ldots > Q(T) \tag{17}$$

The former inequality follows directly from

$$s_{2k+2} = \int_{-1}^{1} z^2 z^{2k} g(z_T) dz_T$$

$$\leq \int_{-1}^{1} z^{2k} g(z_T) dz_T$$

$$= s_{2k}$$

with equality if, and only if,

$$g(z_T) = [1 - Q(T)][\delta(z_T-1) + \delta(z_T+1)] \tag{18}$$

Since a term of the form (18) has already been subtracted from $p(z_T)$, equality is not possible and therefore

$$s_{2k+2} < s_{2k}$$

which, together with (11), establishes (17). Although we have not succeeded in obtaining a precise mathematical formulation, it seems reasonable to conjecture that, for most random processes of interest,

$$\lim_{k \to \infty} s_{2k} = 0$$

and therefore,

$$\lim_{k \to \infty} q_{2k} = Q(T)$$

In light of (17), it appears that (12) should be evaluated for as large a value of k as possible. For the special case when $x(t)$ is a stationary Gaussian random process (assumed to be zero-mean without loss of generality), it does not seem to be possible to evaluate q_{2k} for $k > 1$ as evidenced by the following discussion.

As shown by McFadden in [3] the quantity $R(t_1,t_2,\ldots,t_n)$ [defined in (15)], is equal to the sum of some simple terms plus a quantity $P_n(\vec{r})$, which is defined as

$$P_n(\vec{r}) = (2\pi)^{-n/2}|\vec{r}|^{-\frac{1}{2}} \int_0^\infty dx_1 \ldots \int_0^\infty dx_n \exp\left[-\frac{1}{2}\sum_{i,j}^{n} r_{ij}^{-1}x_i x_j\right] \qquad (19)$$

where $\vec{r}$ is a covariance matrix with elements

$$r_{ij} = r(t_i - t_j) = E\{x(t_i)x(t_j)\}, \quad i,j = 1,\ldots,n$$

$|\vec{r}|$ is the determinant of $\vec{r}$

$\sum_{i,j}^{n} r_{ij}^{-1}x_i x_j$ is the quadratic form associated with the inverse of $\vec{r}$

and

$$x_i = x(t_i), \quad i = 1,2,\ldots,n$$

In other words, $P_n(\vec{r})$ is the probability that the n jointly distributed Gaussian random variables, $x(t_i)$ $(i = 1,\ldots,n)$ are all positive.

As discussed in [3], and more thoroughly in [7], expressions for $P_n(\vec{r})$ have not been obtained in terms of elementary functions for $n > 3$. Because of this fact, it seems unlikely that an expression for (14) with $k > 1$ can be obtained for a general Gaussian process, $x(t)$. It should be pointed out, however, that an expression for q_4 has been derived [8] for $\rho(\tau) = \exp(-|\tau|)$, but without first evaluating $P_4(\vec{r})$. This result is not included because, for this correlation function $Q(T)$ is known exactly [7].

APPLICATION TO GAUSSIAN NOISE

Assume that, $x(t)$, in (1) is a stationary, zero-mean, Gaussian random process, normalized so that $\rho(0) = 1$ where $\rho(\tau) = E\{x(t)x(t+\tau)\}$. The relationship (12) will be evaluated for the case $k = 1$, i.e., for

$$q_2 = \frac{1}{T^2}\int_0^T \int_0^T R(t_1,t_2)dt_1 dt_2 \qquad (20)$$

where

$$R(t_1,t_2) = E\{\operatorname{sgn} x(t_1)\operatorname{sgn} x(t_2)\} \qquad (21)$$

The latter expression has been evaluated by many authors (see p. 58 of [2], for example) and the result is

$$R(t_1,t_2) = \frac{2}{\pi}\sin^{-1}[\rho(t_1 - t_2)] \qquad (22)$$

Substituting (22) into (20) and making the obvious simplifications in integration results in

$$q_2 = \frac{4}{\pi} \int_0^1 (1-u)\sin^{-1}[\rho(Tu)]du,$$

or, in light of (12),

$$Q(T) \leq \frac{4}{\pi} \int_0^1 (1-u)\sin^{-1}[\rho(Tu)]du. \tag{23}$$

This result has been obtained by Slepian [7] where* it is stated as Theorem 5. Slepian's proof, however, is long and complicated, as opposed to the simplicity of the proof given here. Furthermore, extension to other cases can be obtained using the new method.

APPLICATION TO SINE WAVE PLUS GAUSSIAN NOISE

We now turn to applying (12) with $k = 1$ to the case where

$$x(t) = w(t) + A\cos(2\pi ft+\varphi) \tag{24}$$

where

$w(t)$ is a stationary, zero-mean, Gaussian random process with normalized correlation function, $\rho(\tau)$,

φ is a random phase constant uniformly distributed on $[0,2\pi]$,

f is the sine wave frequency,

A is the sine wave amplitude.

Evaluating (14) in this case is lengthy and the derivation will be omitted (see [8] for details). The final result is

$$Q(T) \leq q_2^{(1)} + q_2^{(2)} \tag{25}$$

where

$$q_2^{(2)} = 2\int_0^1 (1-u)H(uT)du \tag{26}$$

---

[*] Note that Slepian's $P[T,r(\tau)]$ equals one-half of our $Q(T)$.

with

$$H(uT) = \frac{2}{\pi} \int_0^{\sin^{-1}\rho(uT)} \exp\left\{- \frac{A^2}{2} \frac{1 - \sin\theta \cos 2\pi bu}{\cos^2\theta}\right\} \times I_o\left\{\frac{A^2}{2} \frac{\sin\theta - \cos 2\pi bu}{\cos^2\theta}\right\} d\theta \tag{27}$$

$$b = fT$$

and

$I_o(x)$ = modified Bessel function of the first kind, zero order.

The expression given below for $q_2^{(1)}$ is approximate, except for $T = k/f$, $k = 0,1,2,\ldots$ where it is exact and consequently the function is most accurate in a neighborhood of these points. In addition, the accuracy of the approximation improves as A increases. For small A (where the approximation is least accurate), (26) dominates $q_2^{(1)}$ and so very little error results in the upper bound, (25). The expression for $q_2^{(1)}$ is given by

$$q_2^{(1)} \simeq \frac{2}{b^2} \times \begin{cases} \displaystyle\int_0^{b-n} (b-n-v)S(v)dv, & n \leq b < n + \tfrac{1}{4} \\[2ex] \displaystyle -\int_{n+\frac{1}{2}-b}^{\frac{1}{4}} (b-n-\tfrac{1}{2}+v)S(v)dv + \int_0^{\frac{1}{4}} (b-n-v)S(v)dv, & n + \tfrac{1}{4} \leq b < n + \tfrac{1}{2} \\[2ex] \displaystyle -\int_0^{b-n-\frac{1}{2}} (b-n-\tfrac{1}{2}-v)S(v)dv + 2\int_0^{\frac{1}{4}} (\tfrac{1}{4}-v)S(v)dv, & n + \tfrac{1}{2} \leq b < n + \tfrac{3}{4} \\[2ex] \displaystyle \int_{n+1-b}^{\frac{1}{4}} (b-n-1+v)S(v)dv - \int_0^{\frac{1}{4}} (b-n-1+v)S(v)dv, & n + \tfrac{3}{4} \leq b < n + 1 \end{cases} \tag{28}$$

where

$$n = 0,1,2,\ldots$$

$$S(v) = (1-4v) - \frac{e^{-A^2/2}}{\pi} G(v) \tag{29}$$

$$G(v) = \int_0^{\pi-2\pi v} e^{-K(v)\cos x} dx - \int_0^{2\pi v} e^{K(v)\cos x} dx \tag{30}$$

and

$$K(v) = \frac{A^2}{2} \cos 2\pi v \tag{31}$$

While (28) appears to be a formidable equation, it turns out to be easily computed, partly because S(v) does not depend on b and hence needs only to be computed once for each value of A. The expression (26), on the other hand, turns out to be time consuming to compute, particularly for large values of b.

To illustrate the usefulness of (25), calculations were made for comparison with the previously reported approximate results of Cobb.[1] The quantity derived in [1] is an approximate expression for the p.d.f. of zero-crossing intervals, denoted by $P_o(T)$. The relationship between Q(T) and $P_o(T)$ is given in [6] as

$$Q(T) = 1 - 2\nu T + 2\nu \int_0^T \int_0^x P_o(t)dt\, dx \tag{32}$$

where ν = expected number of zero-crossings of (24).

As observed in Figs. 1 and 2 of [1],

$$\nu \cong f$$

for the large sine wave amplitudes of interest. Thus,

$$Q(T) \cong 1 - 2fT + \int_0^{2fT} \int_0^y P_o(s)ds\, dy \tag{33}$$

Cobb shows in Eq. (52) of [1] that

$$P_o(s) \cong \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{(s-1)^2}{\sigma^2}\right] \tag{34}$$

where

$$\sigma = \frac{\sqrt{2(1+\rho_1)}}{\pi A}$$

$$\rho_1 = \rho(2fT)$$

The approximation (34) is only valid for $\sigma \ll 1$.

Substituting (34) into (33), we obtain

$$Q(T) \cong \left(1-2fT\right)\left[.5 + \mathrm{erf}\left(\frac{1-2fT}{\sigma}\right)\right] + \frac{\sigma}{\sqrt{2\pi}}\left[e^{-\frac{1}{2}\left(\frac{1-2fT}{\sigma}\right)^2} - e^{-\frac{1}{2\sigma^2}}\right] \qquad (35)$$

As in [1], set

$$\rho(\tau) = \frac{\sin \tau}{\tau} \qquad (36)$$

$$A = 3$$

$$2\pi f = .875 \text{ rad/sec} \qquad (38)$$

The table given below compares the approximate solution based on Cobb's results, (35), and our upper bound, (25). For $2fT < 1$, the approximate solution is somewhat smaller than the upper bound. For $2fT > 1$, the approximation becomes negative and therefore of little interest while the upper bound gradually approaches zero as T increases.

COMPARISON OF APPROXIMATE SOLUTION AND UPPER BOUND ON Q(T) UNDER CONDITIONS (36)-(38)

2fT	Upper Bound	Approximate Solution
0	1	.972
.2	.924	.772
.4	.797	.572
.6	.653	.372
.8	.508	.176
1.0	.370	.028
1.2	.248	-.024
1.4	.152	-.028
1.6	.088	-.028

EXTENSIONS TO OTHER CASES

The specific applications discussed should not be considered exhaustive. For example, the case where x(t) is the sum of a sine wave plus Gaussian noise could easily be extended to x(t) being the sum of a square wave plus Gaussian noise. Although the specific formulae may be more complex, the general result [(12) and (14)] is still applicable for x(t) nonstationary or non-Gaussian.

In addition to the direct extensions just mentioned, the derivation of the general result can be modified slightly to obtain a useful upper bound on the conditional probability that x(t) does not cross the zero axis for an interval of length T, given that x(t) = 0 at the start of the interval. This latter probability has also been intensively investigated (see [7] for discussion).

ACKNOWLEDGMENT

The research reported herein was sponsored in part by the Doctoral Support Plan of Bell Telephone Laboratories, Incorporated; and in part by the National Science Foundation under grant GK-1075. Mr. N. A. Strakhov is on assignment from Bell Telephone Laboratories at Murray Hill, New Jersey.

The paper is based upon a portion of a dissertation submitted to the Graduate Division of The School of Engineering and Science of New York University in partial fulfillment of the requirements for the Ph.D. degree.

REFERENCES

[1] S. M. Cobb, "The Distribution of Intervals Between Zero Crossings of Sine Wave Plus Random Noise and Allied Topics," IEEE Transactions on Information Theory, Vol. IT-11, April, 1965, pp. 220-231.

[2] J. L. Lawson and G. E. Uhlenbeck, "Threshold Signals," McGraw-Hill Book Co., Inc., N. Y., 1950.

[3] J. A. McFadden, "Urn Models of Correlation and a Comparison with the Multivariate Normal Integral," Ann. Math. Stat., Vol. 26, September, 1955, pp. 478-489.

[4] G. F. Newell and M. Rosenblatt, "Zero-Crossing Probabilities for Gaussian Stationary Processes," Ann. Math. Stat., Vol. 33, December, 1962, pp. 1306-1313.

[5] S. O. Rice, "Mathematical Analysis of Random Noise," The Bell System Technical Journal, Vol. 23, 1944, pp. 282-332; Vol. 24, 1945, pp. 46-156. See Esp. Section 3.3, 3.4.

[6] S. O. Rice, "Distribution of the Duration of Fades in Radio Transmission," The Bell System Technical Journal, Vol. 37, May, 1958, pp. 581-636.

[7] D. Slepian, "The One-Sides Barrier Problem for Gaussian Noise," The Bell System Technical Journal, Vol. 41, March, 1962, pp. 463-501.

[8] N. A. Strakhov, "A Representation Theory for a Class of Non-Gaussian Distributions with Applications to Digital Data Transmission," Doctoral Dissertation, New York University, New York, N. Y., October, 1967.

[9] E. Wong, "Some Results Concerning the Zero-Crossings of Gaussian Noise," SIAM J. Appl. Math., Vol. 14, November, 1966, pp. 1246-1254.

STOCHASTIC APPROXIMATION ALGORITHMS FOR SYSTEM IDENTIFICATION, ESTIMATION, AND DECOMPOSITION OF MIXTURES

G. N. Saridis and K. S. Fu
School of Electrical Engineering
Purdue University
Lafayette, Indiana

Z. J. Nikolic
Esso Product Research
Houston, Texas

ABSTRACT

A stochastic approximation procedure that minimizes a mean-square-error criterion is proposed in this paper. It is applied first to derive an algorithm for recursive estimation of the mean-square-error approximation of the function which relates the input signals and the responses of a memoryless system. The input signals are assumed to be generated at random with an unknown probability density function, and the response is measured with an error which has zero mean and finite variance. A performance index for evaluating the rate of convergence of the algorithm is defined and then the optimal form of the algorithm is derived. It is shown that the least-square-error fit of the measured output signals of the systems offers a recursive formula which is a special case of the proposed algorithm. A recursive formula for estimation of a priori probabilities of the pattern classes using unclassified samples is then presented. The rate of convergence is computed.

A minimum square-error-estimate of a continuous probability density function is also obtained by the same algorithm.

I. INTRODUCTION

A stochastic approximation procedure that minimizes a mean-square-error criterion, and its rate of convergence are studied. This procedure is first applied for identification of unknown systems, required during the design or physical realizations of optimal controllers for such systems. An algorithm for obtaining the mean-square-error approximation of the function which relates the input signals and the responses of a memoryless system is described here. Similar problems were treated recently by several authors [1,2,3,4,5]. The multidimensional stochastic approximation procedures of Blum and Kiefer-Wolfowitz [7] were used in the solution of such problems. They were compared experimentally with least-square-error approximation procedures [3,4].

A performance index for evaluating the rate of convergence of the proposed algorithm is defined and then the optimal form of the algorithm is derived. It is shown that the least-square-error fit of the measured output signals of the systems offers a recursive formula which is a special case of the proposed algorithm.

The second part of the paper is concerned with a problem of learning in statistical pattern recognition system. The quality of operation of a statistical pattern recognition system always depends on the accuracy of the available probabilistic descriptions of the pattern classes [9,10]. Very often the correct classifications of the observed patterns are not available. In that case "learning without a teacher" should take place. Several methods for this type of learning exist [11]. They are usually made feasible by reducing the problem of learning to a process of successive estimation either of some unknown parameters underlying probability distribution function of all pattern classes (mixture distribution) or of a known decision boundary. A recursive formula for successive estimation of a priori probabilities of the pattern classes is derived here for the case when the observed samples are unclassified. The probability density functions of the pattern classes under consideration are assumed to be known. The rate of convergence is computed for the algorithm. A minimum square-error-estimate of a continuous probability density function is also obtained by the same algorithms.

II. STATEMENT OF THE PROBLEM

The memoryless system under consideration (Figure 1) has m input terminals ($m<\infty$) and a single output terminal. An input signal is described by a point in m-dimensional Euclidean space Ω_x. The collection of all the available input signals is represented by the set of points D in Ω_x. Due to physical limitations of the input signals, every $x\in D$ has the norm $||x|| = |(\sum_{i=1}^{m} x_{(i)}^2)^{1/2}| < \infty$, where $x_{(i)}$ is the i-th coordinate of the point x in Ω_x. The sequence of signals applied at the input terminals of the system, $\{x_n : x_n \in D; n=1,2,\ldots\}$, is considered to be an infinite sequence of independent and identically distributed samples with continuous probability density function p(x) such that

$$p(x) > 0 \quad \text{for all } x\in D, \text{ and } \int_D p(x)dx = 1 \tag{1}$$

The response of the memoryless system on the input signal $x\in D$ is

$$z = f(x) \tag{2}$$

where f is a real-valued, continuous and bounded function defined over D. The actual measurable output signal $x\in D$ is

$$y = f(x) + v \tag{3}$$

where v is the measurement noise which is independent of x. It is assumed that y is a random variable with a conditional probability density function p(y/x) such that

$$E[y/x] \triangleq \int_{-\infty}^{+\infty} yp(y|x)dy = f(x) \quad \text{and} \tag{4}$$

$$E[y^2/x] \triangleq \int_{-\infty}^{+\infty} y^2 p(y|x)dy < \infty \quad \text{for every } x\in D. \tag{5}$$

The identification of a memoryless system described by (2) consists in finding an estimate $\hat{f}$ of the unknown function f defined over D. The estimate of f is defined here as

$$\hat{f}(x) = c^T\emptyset(x) = \sum_{i=1}^{N} c_{(i)}\varphi_i(x) \tag{6}$$

where $c^T = (c_{(1)}, c_{(2)}, \ldots, c_{(N)})$ is a row vector with N real components, and $\emptyset(x)$ is a column vector with the components $\varphi_i(x)$, $i = 1,2,\ldots,N$; $N<\infty$ and $x\in D$. The set of functions

$$G = \{\varphi_i(x) : x\in D; i=1,2,\ldots,N; N<\infty\} \tag{7}$$

consists of certain preselected functions which are linearly independent, real-valued, continuous and bounded over D. The vector c^T in (6) is chosen to minimize the mean-square-error criterion

$$I(c^T) \triangleq \frac{1}{2} E_x[(f(x)-c^T\emptyset(x))^2] = \frac{1}{2}\int_D (f(x)-c^T\emptyset(x))^2 p(x)dx. \tag{8}$$

Its components are obtained from the equation

$$\frac{dI(c^T)}{dc^T} = E_x[(f(x)-c^T\emptyset(x))\emptyset(x)] = 0 \tag{9}$$

Since the expectations

$$V \triangleq E_x[f(x)\emptyset(x)] = \int_D f(x)\emptyset(x)p(x)dx \tag{10}$$

and

$$M^{-1} \triangleq E_x[\emptyset(x)\emptyset^T(x)] = \int_D \emptyset(x)\emptyset^T(x)p(x)dx \tag{11}$$

exist, equation (9) can be written as

$$V = M^{-1}c \tag{12}$$

The solution of (12) exists when the inverse of M^{-1} exists, and is equal to

$$c = MV \tag{13}$$

Therefore, in order to obtain the estimate $\hat{f}$ of f as defined by (6) and (12), it is necessary to choose the functions in G so that the determinant of the matrix M^{-1} is

$$\det M^{-1} \neq 0 \tag{14}$$

Since 1) $|f(x)| < \infty$ for every $x \in D$, 2) $|\varphi_i(x)| < \infty$ for every $x \in D$ and every $\varphi_i(x) \in G$, 3) $E[y|x] = f(x)$ and $E[y^2|x] < \infty$ for every $x \in D$, then

$$\begin{aligned} E[y\emptyset(x)] &= \int_D \int_{-\infty}^{+\infty} y\emptyset(x)p(x,y)dy\,dx = \int_D \int_{-\infty}^{+\infty} y\emptyset(x)p(y|x)p(x)dy\,dx \\ &= \int_D \left(\int_{-\infty}^{+\infty} yp(y|x)dy\right)\emptyset(x)p(x)dx = E_x[f(x)\emptyset(x)] = V \end{aligned} \tag{15}$$

and

$$E[y^2\varphi_i^2(x)] \triangleq \int_D \int_{-\infty}^{+\infty} y^2\, \varphi_i^2(x)\, p(y|x)p(x)dy\,dx < \infty \tag{16}$$

for every $\varphi_i(x) \in G$.

The successive estimates of c in (13) can now be formed from the successive estimates of M^{-1} and $E[y\emptyset(x)] \triangleq V$ even if p(x) is not explicitly known. The following section describes an algorithm for successive estimation of c when $M^{-1} \triangleq E_x[\emptyset(x)\emptyset^T(x)]$ is a symmetric NxN positive definite matrix.

III. PROPOSED ALGORITHM FOR IDENTIFICATION OF MEMORYLESS SYSTEMS

The n-th estimates of M^{-1} and V are designated respectively by M_n^{-1} and V_n. Let

$$\emptyset_n \emptyset_n^T \triangleq \emptyset(x_n)\, \emptyset^T(x_n) \tag{17}$$

and

$$y_n \emptyset_n \triangleq y_n \emptyset(x_n) \tag{18}$$

where y_n is the response on the n-th applied input signal x_n.

The n-th estimate of M^{-1} is obtained by the recursive formula

$$M_n^{-1} = (1-\beta_n)M_{n-1}^{-1} + \beta_n \emptyset_n \emptyset_n^T \tag{19}$$

where 1) M_0^{-1} is any symmetric positive definite matrix with bounded elements, and 2) β_n is a real number such that for every $n \geq 1$

$$\beta_n > 0, \ (1-\beta_n) > 0, \ \prod_{n=1}^{\infty} (1-\beta_n) = 0 \text{ and } \sum_{n=1}^{\infty} \beta_n^2 < \infty \tag{20}$$

Equation (19) can be rewritten in the form

$$M_n^{-1} - M^{-1} = (1-\beta_n)(M_{n-1}^{-1} - M^{-1}) + \beta_n H_n \tag{21}$$

where

$$H_n \triangleq \emptyset_n \emptyset_n^T - M^{-1} \tag{22}$$

Since x_n is generated with probability density function p(x), H_n is NxN symmetric matrix of which the entries are the random variables with expectations

$$E_x[\varphi_i(x)\, \varphi_j(x) - E_x[\varphi_i(x)\, \varphi_j(x)]] = 0 \tag{23}$$

and

$$E_x[|\varphi_i(x)\, \varphi_j(x)\, \varphi_k(x)\, \varphi_\ell(x)|] < \infty \tag{24}$$

for every $i,j,k,\ell=1,\ldots,N$ and $n=0,1,2,\ldots$ Therefore, the recursive formula (19) with (20) corresponds to the special case of stochastic approximation [6] for which

$$\text{Prob}\{\lim_{n\to\infty} M_n^{-1} = M^{-1}\} = 1. \tag{25}$$

Since $(1-\beta_1)M_0^{-1}$ is a symmetric positive definite matrix, and $\beta_n\emptyset_n\emptyset_n^T$ is a symmetric positive semi-definite matrix, the matrices M_n^{-1} obtained by (19) are positive definite for every $n > 0$. The existence of M_n is thus guaranteed for every $n > 0$. This property is now used to obtain the recursive formula for computation of M_n. It is known [7] that for two symmetric definite matrices A and B (matrix inversion lemma)

$$A^{-1} = B^{-1} + \alpha\alpha^T \text{ implies } A = B - (1+\alpha^T B\alpha)^{-1} B\alpha\alpha^T B \tag{26}$$

Therefore, the recursive formula (19) is equivalent to

$$M_n = \frac{1}{1-\beta_n}\left[M_{n-1} - \frac{1}{\frac{1-\beta_n}{\beta_n} + \emptyset_n^T M_{n-1} \emptyset_n} M_{n-1}\emptyset_n\emptyset_n^T M_{n-1}\right] \tag{27}$$

If M_0 is any symmetric definite matrix and β_n is defined by (20), then the recursive formula (27) is such that

$$\text{Prob}\{\lim_{n\to\infty} M_n = M\} = 1 \tag{28}$$

where M is a symmetric positive definite matrix.

The n-th estimate of V is defined by

$$V_n = V_{n-1} + \gamma_n [y_n \emptyset_n - V_{n-1}] \tag{29}$$

where 1) V_o is any column vector with N components such that $||V_o|| < \infty$ and 2) γ_n is a real number such that for every n

$$\gamma_n > 0,\ (1-\gamma_n) > 0,\ \prod_{n=1}^{\infty} (1-\gamma_n) = 0 \text{ and } \sum_{n=1} \gamma_n^2 < \infty. \tag{30}$$

The recursive relation (28) can be written as

$$V_n - V = (1-\gamma_n)(V_{n-1} - V) + \gamma_n \xi_n \tag{31}$$

where

$$\xi_n \overset{\Delta}{=} y_n \emptyset_n - V \tag{32}$$

Since ξ_n is a vector of which the components are random variables with expectations $E[y\varphi_i(x) - E[y\varphi_i(x)]] = 0$ and $E[y^2\varphi_i^2(x)] < \infty$ (33) for every $\varphi_i(x) \in G$, the recursive formula (29) and (30) is again a special case of stochastic approximation [6] for which

$$\text{Prob}\{\lim_{n\to} V_n = V\} = 1 \tag{34}$$

Finally, the n-th estimate of c is $c_n = M_n V_n$. The recursive computation of c is performed according to the following formulas:

$$c_n = \frac{1-\gamma_n}{1-\beta_n} c_{n-1} + \frac{M_{n-1}\emptyset_n}{\frac{1-\beta_n}{\beta_n} + \emptyset_n^T M_{n-1}\emptyset_n} \left[\frac{\gamma_n}{\beta_n} y_n - \frac{1-\gamma_n}{1-\beta_n} \emptyset_n^T c_{n-1}\right] \tag{35}$$

and

$$M_n = \frac{1}{1-\beta_n} [M_{n-1} - \frac{1}{\frac{1-\beta_n}{\beta_n} + \emptyset_n^T M_{n-1}\emptyset_n} M_{n-1} \emptyset_n \emptyset_n^T M_{n-1}] \tag{36}$$

where 1) M_o is any symmetric definite matrix with real bounded entries; 2) c_o is any N-dimensional column vector with bounded real components; 3) $\{\gamma_n\}$ is defined by (30); and 4) $\{\beta_n\}$ is defined by (20).

When $\beta_n = \gamma_n$ for every n in (35) and (36),

$$c_n = c_{n-1} + \frac{M_{n-1} \emptyset_n}{\frac{1-\gamma_n}{\gamma_n} + \emptyset_n^T M_{n-1}\emptyset_n} [y_n - c_{n-1}^T \emptyset_n] \tag{37}$$

and

$$M_n = \frac{1}{1-\gamma_n} [M_{n-1} - \frac{1}{\frac{1-\gamma_n}{\gamma_n} + \emptyset_n^T M_{n-1} \emptyset_n} M_{n-1} \emptyset_n \emptyset_n^T M_{n-1}]. \tag{38}$$

Given from (28) and (34) that $\{M_n\} \xrightarrow{W.P.1} M$ and $\{V_n\} \xrightarrow{W.P.1} V$, the identity Prob(A+B)=Prob(A)+Prob(B)-Prob(AB) implies that $\text{Prob}\{M_n \to M, V_n \to V\} = 1$. Therefore

$$\text{Prob}\{\lim_{n\to} M_n V_n = c\} = 1 \tag{39}$$

where c = MV, and c_n converges to c W.P.1.

VI. A COMPARISON WITH THE LEAST-SQUARE-ERROR ALGORITHMS

The method of least-square-error [12] was used in the solution of a similar problem in [4]. The least-square-error fit of the successively measured output signals $y_1, y_2, \cdots, y_n$ with the function $\hat{f}(x) = c^T\emptyset(x) = \emptyset^T(x)c$ is first briefly described and then compared with the algorithm (35)-(36).

Let y_k be the output signal of the system on the input signal x_k, and let

$$\emptyset_k^T \overset{\Delta}{=} \emptyset^T(x_k) = (\varphi_1(x_k), \varphi_2(x_k), \cdots, \varphi_N(x_k)). \tag{40}$$

It is desired to find c_n which minimizes the norm

$I(c_n) = ||Y_n - \psi_n c_n||^2$ where

$$Y_n = [y_1, y_2, \ldots, y_n]^T \quad \text{and} \quad \psi_n = [\phi_1, \phi_2, \ldots, \phi_n]^T \tag{41}$$

$I(c_n)$ is minimized when

$$\frac{dI(c_n)}{dc_n} = 0 \quad \text{or} \quad (Y_n - \psi_n c_n)\psi_n = 0 \tag{42}$$

The solution of (42) exists only when the inverse of $\psi_n^T \psi_n$ exists, and is equal to

$$c_n = [\psi_n^T \ \psi_n]^{-1} \ \psi_n Y_n \tag{43}$$

Let

$$M_n^{-1} = \frac{1}{n} \psi_n^T \ \psi_n = \frac{1}{n} \sum_{k=1}^{n} \phi_k \ \phi_k^T \tag{44}$$

Since

$$\psi_n^T \ \psi_n = \psi_{n-1}^T \ \psi_{n-1} + \phi_n \ \phi_n^T \tag{45}$$

and

$$\psi_n^T \ Y_n = \psi_{n-1}^T \ Y_{n-1} + \phi_n \ y_n \tag{46}$$

then the n-th least-square-estimate is from (43)

$$c_n = [n \ M_n^{-1}]^{-1} \ [\psi_{n-1} \ Y_{n-1} + \phi_n \ y_n] \tag{47}$$

or

$$c_n = c_{n-1} + \frac{1}{n} M_n \ \phi_n [y_n - c_{n-1}^T \phi_n], \tag{48}$$

where M_n is the inverse of

$$M_n^{-1} = \frac{n-1}{n} \ M_{n-1}^{-1} + \frac{1}{n} \phi_n \ \phi_n^T \tag{49}$$

If for certain $n^* \geq N$ the matrix $M_{n^*}^{-1}$ is a positive definite matrix, then M_n^{-1} is a positive definite matrix for $n > n^*$ since $\phi_n \phi_n^T$ is a positive semi-definite matrix. Therefore, by applying (26) to (49),

$$M_n = \frac{n}{n-1} [M_{n-1} - \frac{1}{n-1 + \phi_n^T \ M_{n-1} \phi_n} M_{n-1} \ \phi_n \ \phi_n^T \ M_{n-1}] \tag{50}$$

By substituting (50) into (48),

$$c_n = c_{n-1} + \frac{M_{n-1} \ \phi_n}{n-1 + \phi_n^T \ M_{n-1} \ \phi_n} [y_n - c_{n-1}^T \ \phi_n] \ . \tag{51}$$

The recursive formula (51) with (50) is a special case of the recursive formula (35) with (36) when 1) $\beta_n \equiv \gamma_n = \frac{1}{n}$ for every $n > 1$ and 2) M_n^{-1} is a positive definite symmetric matrix starting from a certain $n \geq N$. Therefore, the sequence $\{c_n\}$ obtained by the least-square-error algorithm (51) with (50) converges to $c = MV$ with probability one as it is indicated by (39).

It is interesting to notice that the recursive formula (51) is also obtained by computing $c = M \ V$ where M and V are obtained as sample means, i.e., M_n as defined by (50) and

$$V_n = \frac{1}{n} \sum_{k=1}^{n} y_k \ \phi_k \ . \tag{52}$$

V. ACCELERATION OF CONVERGENCE

The rate of convergence of the proposed algorithm (35) with (36) will be evaluated by the expectation

$$J_n \overset{\Delta}{=} \frac{1}{2} E[||c_n - c||_A^2] \tag{53}$$

where the norm

$$||c_n - c||_A^2 \overset{\Delta}{=} (c_n - c)^T \ M^{-1} \ M^{-1} (c_n - c) \tag{54}$$

and M^{-1} is the positive definite matrix defined by (11) and A is defined accordingly. The acceleration of the algorithm is performed by selecting the sequences $\{\gamma_n\}$ in (30) and $\{\beta_n\}$ in (20) so that an upper bound on J_n is minimized for every $n \geq 1$.

From (12) and $M_n^{-1} c_n = V_n$,

$$M_n^{-1} c_n - M^{-1} C = V_n - V \tag{55}$$

or

$$M^{-1}(c_n - c) + (M_n^{-1} - M^{-1})c_n = V_n - V \tag{56}$$

Then, the euclidean norm

$$||M^{-1}(c_n-c)|| \leq ||V_n-V|| + ||(M_n^{-1} - M^{-1})c_n|| \tag{57}$$

or

$$||c_n-c||_A^2 = ||M^{-1}(c_n-c)||^2 \leq 2||V_n-V||^2 + 2||(M_n^{-1} - M^{-1})c_n||^2 \tag{58}$$

$$\leq 2||V_n-V||^2 + 2|(M_n^{-1} - M^{-1})^2| \; ||c_n||^2 \leq 2||V_n-V||^2 + 2|(M_n^{-1}-M^{-1})^2|b_{n_2}^2$$

where $|(M_n^{-1}-M^{-1})^2|$ is the norm of the matrix $(M_n^{-1}-M^{-1})^2$, and $||c_n||^2 < b_n^2$. By taking the expected values in (58),

$$J_n \leq E[||V_n-V||^2] + E[|(M_n^{-1} - M^{-1})^2|]b_n^2 \leq B_n^{(V)} + B_n^{(M)} b_n^2 = B_n \tag{59}$$

where

$$E[||V_n-V||^2] \leq B_n^{(V)} \quad \text{and} \quad E[|(M_n^{-1} - M^{-1})^2|] \leq B_n^{(V)} \tag{60}$$

An upper bound $B_n^{(V)}$ will be now estimated. From (31) and (32),

$$||V_n-V||^2 = (1-\gamma_n)^2||V_{n-1}-V||^2 + (1-\gamma_n)\xi^T(V_{n-1}-V) \tag{61}$$

$$+ (1-\gamma_n)(V_{n-1}-V)^T\xi_n + \gamma_n^2||\xi_n||^2$$

and due to (33)

$$E[||V_n-V||^2] \leq (1-\gamma_n)^2 E[||V_{n-1}-V||^2] + \gamma_n^2 \sigma^2 \tag{62}$$

where $E[||y\phi(x)-V||^2] \leq \sigma^2 < \infty$ for every n. Then

$$B_n^{(V)} = (1-\gamma_n)^2 B_{n-1}^{(V)} + \gamma_n^2 \sigma^2 \tag{63}$$

Similarly, from (21) and (23),

$$E[|(M_n^{-1}-M^{-1})^2|] \leq (1-\beta_n)^2 E[|(M_{n-1}^{-1} - M^{-1})^2|] + \beta_n^2 E[|H_n|^2] \tag{64}$$

and

$$B_n^{(M)} = (1-\beta_n)^2 B_{n-1}^{(M)} + \beta_n^2 \tau^2 \tag{65}$$

where

$$E[|\phi\phi^T - M^{-1}|^2] \leq \tau^2 < \infty. \tag{66}$$

B_n in (59) is minimized when $B_n^{(V)}$ and $B_n^{(M)}$ are minimized at each stage n. From $dB_n/d\gamma_n = 0$,

$$\gamma_n = \frac{B_{n-1}^{(V)}}{\sigma^2 + B_{n-1}^{(V)}} \quad \text{and} \quad B_n^{(V)} = \frac{B_{n-1}^{(V)} \sigma^2}{\sigma^2 + B_{n-1}^{(V)}} . \tag{67}$$

From $dB_n/d\beta_n = 0$,

$$\beta_n = \frac{B_{n-1}^{(M)}}{\tau^2 + B_{n-1}^{(M)}} \quad \text{and} \quad B_n^{(M)} = \frac{B_{n-1}^{(M)} \tau^2}{\tau^2 + B_{n-1}^{(M)}} \tag{68}$$

By iterating (67) and (68),

$$\gamma_n = 1/(n + \frac{\sigma^2}{B_0^{(V)}}) \text{ with } B_n^{(V)} = \sigma^2/(n + \frac{\sigma^2}{B_0^{(V)}}) , \tag{69}$$

and

$$\beta_n = 1/(n + \frac{\tau^2}{B_0^{(M)}}) \text{ with } B_n^{(M)} = \tau^2/(n + \frac{\tau^2}{B_0^{(M)}}) \tag{70}$$

where

$$E[||V_0-V||^2] \leq B_0^{(V)} < \infty \quad \text{and} \quad E[||M_0^{-1}-M^{-1}||^2] \leq B_0^{(M)} < \infty \tag{71}$$

The sequences $\{\gamma_n\}$ and $\{\beta_n\}$ defined in (69) and (70) have the properties described in (30) and (20) respectively. Therefore, the least-square-error algorithm, for which $\gamma_n = \frac{1}{n}$ for $n > 1$, is not optimal with respect to the performance index (53).

It is noted that if (55) is written as $M_n^{-1}(c_n-c)+(M_n^{-1}-M^{-1})c = V_n-V$, then $||M_n^{-1}(c_n-c)|| \leq ||V_n-V|| + ||(M_n^{-1}-M^{-1})c||$. By following (57)-(59), $E[||c_n-c||^2 M_n^{-1} - M_n^{-1}] \leq B_n^{(V)} + B_n^{(M)} b^2 = B_n$ where $||c||^2 \leq b^2 <$, and $B_n^{(V)}$ and $B_n^{(M)}$ are defined in (69) and (70). Since $\lim_{n\to\infty} B_n^{(V)} = 0$ and $\lim_{n\to\infty} B_n^{(M)} = 0$, then $\lim_{n\to\infty} E[||c_n-c||^2{}_{M_n^{-1}} {}_{M_n^{-1}}] = 0$.

VI. DECOMPOSITION OF MIXTURES

This section is concerned with a problem of "learning without a teacher" in statistical pattern recognition systems [10,11]. It is assumed that the set of pattern classes under consideration is finite and designated by $\Omega_c \triangleq \{\omega_i : I\ i = 1,2,\ldots,A; A < \infty\}$. The feature vector x describes a pattern from Ω_c and belongs to an m-dimensional Euclidean space $\Omega_x; m < \infty$. The probability density function of the patterns which belong to the pattern class ω_i is $f_i(x)$, and the mixture of Ω_c defined over Ω_x is

$$p(x) = \sum_{i=1}^{A} g_i\, f_i(x) \tag{72}$$

where g_i is the a priori probability of the pattern class ω_i, $i=1,2,\ldots,A$, and $\sum_{i=1}^{A} g_i = 1$.

A recursive formula is derived for the estimation of the a priori probabilities g_i, $i=1,2,\ldots,A$, in the mixture p(x) when $f_i(x)$ is known for every i. The values $\hat{g}_i$, $i=1,2,\ldots,A$, which minimize

$$I(\hat{g}) = \int_{\Omega_x} \Big(\sum_{i=1}^{A} (\hat{g}_i - g_i) f_i(x)\Big)^2 dx + 2\lambda\Big(\sum_{i=1}^{A} \hat{g}_i - 1\Big) \tag{73}$$

where λ is a Lagrange multiplier, are obtained by solving simultaneously the system of linear equations obtained by setting $\partial I(\hat{g})/\partial \hat{g}_i = 0$ for $i=1,2,\ldots,A$:

$$W\hat{g} = E[\underline{f}(x)] - \lambda\, U \tag{74}$$

where (1) W is the matrix with elements $\omega_{ij} = \int_{\Omega_x} f_i(x) f_j(x)\, dx$, $i,j=1,2,\ldots,A$, (2) $E[\underline{f}(x)]$ is a column vector with the i-th component equal to $E[f_i(x)] \triangleq \int_{\Omega_x} f_i(\bar{x})\, p(x)\, dx$, $i=1,2,\ldots,A$ and (3) U is a column vector of A components all equal to 1. It is required that det W, the determinant of W, is not equal to zero. The i-th component of $\hat{g}$ which satisfies (74) is then

$$\hat{g}_i = \sum_{k=1}^{A} \left(E[f_k(x)] - \frac{\sum_{\ell=1}^{A} \sum_{j=1}^{A} E[f_j(x)] W_{j\ell} - \det W}{\sum_{\ell=1}^{A} \sum_{j=1}^{A} W_{j\ell}}\right) \frac{W_{ki}}{\det W} \tag{75}$$

where $W_{j\ell}$ is the adjunct of $\omega_{j\ell}$ in the matrix W. Since equation (74) has a unique solution, and $A < \infty$, then $\hat{g}_i = E[F_i(x)]$ where

$$F_i(x) = \sum_{k=1}^{A} \left\{f_x(x) - \frac{\sum_{\ell=1}^{A} \sum_{j=1}^{A} f_j(x) W_{j\ell} - \det W}{\sum_{\ell=1}^{A} \sum_{j=1}^{A} W_{j\ell}}\right\} \frac{W_{ki}}{\det W} \tag{76}$$

The a priori probability g_i can now be estimated by using n independent unclassified samples $x_1, x_2, \ldots, x_n$ from the mixture $p(x) = \sum_{i=1}^{A} g_i f_i(x)$. The following recursive formula for estimation of g_i is suggested: the n-th estimate of g_i is

$$g_i(n) = g_i(n-1) + \gamma_n [F_i(x_n) - g_i(n-1)];\ i=1,2,\ldots,A \tag{77}$$

where 1) $F_i(x)$ is defined by (76),](2) γ_n is defined by (30) for every $i=1,2,\ldots,A$, and 3) $g_i(0) \geq 0$ with $\sum_{i=1}^{A} g_i(0) = 1$.

The algorithm (77) with (30) belongs to the special case of stochastic approximation [6] for which

$$P\{\lim_{n\to\infty} g(n) = g\} = 1 \quad \text{and} \quad \lim_{n\to\infty} E[||g(n)-g||^2] = 0 \tag{78}$$

where $g = (g_1, g_2, \ldots, g_A)$ and $g(n) = (g_1(n), g_2(n), \ldots, g_A(n))$. A proof of (78) is given here for a particular sequence $\{\gamma_n\}$ of (30).

The expected value of the norm $||g(n)-g||^2 = g(n)-g)^T(g(n)-g)$ is from (77) and (76): $E[||g(n)-g||^2] = (1-\gamma_n)^2 E[||g(n-1)-g||^2] + \gamma_n^2\, E||\xi(n)||^2$ (79)
where $\xi_i(n)$ is A-dimensional row vector with the components

$$\xi_i(n) = \left(\sum_{k=1}^{A}\left\{f_k(x_n) - \frac{\sum_{\ell=1}^{A}\sum_{j=1}^{A} f_j(x_n)W_{j\ell} - \det W}{\sum_{\ell=1}^{A}\sum_{j=1}^{A} W_{j\ell}}\right\}\frac{W_{ki}}{\det W}\right) - g_i;\ i=1,2,\ldots,A \quad (80)$$

The i-th component of $\xi(n)$ is the random variable $\{F_i(x)-g_i\}$ which has mean zero and variance $E[(F_i(x)-g_i)^2] < \sigma_i^2;\ i=1,2,\ldots,A$ (81)
Let $\sum_{i=1}^{A} \sigma_i^2 = \sigma^2$ and $E[||g(n-1)-g||^2] \leq V_{n-1}^2$, then from (79)

$$0 \leq E[||g(n)-g||^2] \leq V_n = (1-\gamma_n)^2 V_{n-1}^2 + \gamma_n^2 \frac{\sigma^2}{n}. \quad (82)$$

By iterating (82) for $E[||g(0)-g||^2] \leq V_o^2$, the minimal value of V_n^2 is

$$V_n^2 = \sigma^2/(n + \frac{\sigma^2}{V_o^2}) \quad (83)$$

for

$$\gamma_n = 1/(n + \frac{\sigma^2}{V_o^2}) \quad (84)$$

and $E[||g(0)-g||^2] \leq V_o^2$.

VII. ESTIMATION OF CONTINUOUS PROBABILITY DENSITY FUNCTIONS

When the mixture defined in (72) is continuous and bounded over Ω_x, then a linear combination of continuous, bounded and orthonormal functions $\{\varphi_i(x); x\in\Omega_x, |\varphi_i(x)| < \infty, \int_{\Omega_x}\varphi_i(x)\varphi_j(x)dx=\delta_{ij};\ i,j=1,2,\ldots,n;\ n \leq \infty\}$ can be used to represent p(x). The quality of the approximating function $\hat{p}(x) = \sum_{i=1}^{n} g_i\varphi_i(x)$ is evaluated by the integral square error criterion

$$I(g) = \int_{\Omega_x} (p(x) - \sum_{i=1}^{n} g_i\ \varphi_i(x))^2 dx \quad (85)$$

The minimum of I(g) with respect to $g=(g_1, g_2, \ldots, g_n)$ is obtained for

$$g_i = E_x[\varphi_i(x)] = \int_{\Omega_x} \varphi_i(x)p(x)dx;\quad i=1,2,\ldots,n. \quad (86)$$

Successive approximation of g_i can now be performed using independent samples from p(x) according to the algorithm (77) where: 1) $F_i(x) \triangleq \varphi_i(x)$ for $x\in\Omega_x$; 2) $_n\gamma_n$ is defined by (84) with $V_o^2 \geq E[||g(0)-g||^2]=E[\sum_{i=1}^{n}(g_i(0)-g_i)^2]$ and 3) $\sigma^2 \geq E_x[\sum_{i=1}^{n}(\varphi_i(x)-g_i)^2];\ \sigma^2 < \infty$. (87)
As in the case of algorithm (77), it can be shown that (78), for the continuous probability density functions, is true.

VIII. SIMULATION RESULTS

The algorithm (35) with (36) where γ_n and β_n are defined by (67) and (68) was compared with the least-square-error algorithm for which $\beta_n \equiv \gamma_n \equiv \frac{1}{n}$. The memoryless plant is described by $z = f(x) = 2 + 3x^2$ and $y = 2 + 3x^2 + v$ where 1) v is gaussian noise with mean equal to zero and variance equal to 0.0025, and 2) x is uniformly distributed over the interval [-1,+1]. The estimate of f(x) is $\hat{f}(x) = c_1\varphi_1(x) + c_2\varphi_2(x) + c_3\varphi_3(x)$ where $\varphi_1(x) = 1$, $\varphi_2(x) = x$, and $\varphi_3(x) = x^2$. The successive estimates of c_1, c_2 and c_3 are presented in Fig. 2. The curves are obtained by averaging the results of 20 experiments with different random sequences of input signals, and for $\gamma_n = \beta_n$, $\frac{\sigma^2}{B_o(V)} = \frac{\tau^2}{B_o(M)} = .05$ and $O(\frac{1}{n}$ sequence). The initial estimate of M^{-1} is obtained from the first 3 input signals in each of the experiments. The obtained estimates M_3^{-1} were positive definite in all performed experiments. A sequence of 1000 input signals was applied in a single experiment. The effect of the acceleration over the algorithm with $\gamma_n = \frac{1}{n}$ is obvious from the plots.

The algorithms (77) and (84), (algorithm 1) is used for estimating the unknown a priori probability $g_1 = 0.3$ in the mixture of two classes ω_1 and

and ω_2, $p(x)=g_1f_1(x) + g_2f_2(x)$, where 1) $g_1 + g_2 = 1$;

$$2)\quad f_1(x) = \begin{cases} 1/2 \text{ if } -1 < x < 1 \\ 0 \quad \text{otherwise} \end{cases};$$

$$3)\quad f_2(x) = \begin{cases} 1/2 \text{ if } 0 < x < 2 \\ 0 \quad \text{otherwise} \end{cases};$$

The determinant $\text{Det } W = \begin{vmatrix} 1/2 & 1/2 \\ 1/4 & 1/2 \end{vmatrix} = \frac{3}{16}$ and $W_{11} = 1/2$, $W_{12} = 1/4$, $W_{21} = -1/4$, $W_{22} = 1/2$. The successive estimates of g_1 are then computed by $g_1(n)=g_1(n-1)+\gamma_n[2f_1(x_n)-2f_2(x_n)+\frac{1}{2} - g_1(n-1)]$ with (87). The frequency ratio of occurrence of class ω_1 (when the classification of each observed sample was known) is also presented in Figure 3. One experiment consisted of a sequence of 10,000 samples from the mixture. The curves $g_1(n)$ vs. n in Fig. 3 were obtained in a single experiment.

The experiments were simulated on IBM 7094 computer.

IX. CONCLUSIONS

A stochastic approximation procedure was used to identify memoryless systems from the measurements of the output signals corrupted with noise as well as to obtain decomposition of mixtures of probability density functions of pattern classes. The algorithm developed for the identification problem was shown to include the least squares approach as a special case and an accelerated scheme was presented. A recursive formula for estimation of a priori probabilities of the pattern classes utilizing unclassified patterns and estimation of continuous probability density functions is presented. The rate of convergence is also computed. Illustrative examples were used to demonstrate the applications of the various algorithms.

ACKNOWLEDGEMENTS

This research was supported by NASA Institutional Grant NGR 15-005-021 and National Science Foundation Grant GK-696. The authors are grateful to Mr. Gunter Stein for his assistance in programming the examples.

REFERENCES

1. Aizerman, M. A., Braverman, E. M. and Rozonoer, L. I., "Method of Potential Functions in the Problem of Restoration of a Functional Converter Characteristic by Means of Points Observed Randomly", Avtomatika i Telemekhanika, Vol. 25, No. 12, December, 1964. (English translation: Automation and Remote Control)
2. Braverman, E. M. and Pyatnitskii, E. S., "Estimation of the Rate of Convergence for the Algorithms Based on the Method of Potential Functions", Avtomatika i Telemekhanika, No. 1, January, 1966.
3. Kashyap, R. L. and Blaydon, C. C., "Recovery of Functions from Noisy Measurements Taken at Randomly Selected Points and Its Applications to Pattern Recognition", Proc. of the IEEE, Vol. 54, No. 8, August, 1966.
4. Blaydon, C. C. and Ho, Y. C., "On the Abstraction Problem in Pattern Classification", Proc. of NEC, pp. 857-862, Vol. 22, 1966.
5. Tsypkin, Ya. Z., "Adaptation, Learning and Self-Learning in Automatic Systems", Avtomatika i Telemekhanika, No. 1, pp. 23-61, January, 1966.
6. Dvoretzky, A., "On Stochastic Approximation", Proc. of the Third Berekeley Symposium on Math. Stat. and Prob., University of California Press, Los Angeles, Vol. 1, pp. 39-59, 1956.

7. Schmetterer, L., "Stochastic Approximation", Proc. of the Fourth Berkeley Symposium on Math. Stat. and Prob., University of California Press, Los Angeles, Vol. 1, 1958.

8. Fisz, M., Probability Theory and Mathematical Statistics, John Wiley and Sons, Inc., New York, 1963.

9. Nikolic, Z. J. and Fu, K. S., "A Mathematical Model of Learning in an Unknown Random Environment", Proc. of the NEC, Vol. 22, 1966.

10. Braverman, D. and Abramson, N., "Learning to Recognize Patterns in a Random Environment", IRE Trans. on Information Theory, Vol. IT-8, 1965.

11. Spragins, J., "Learning Without a Teacher", IEEE Trans. on Information Theory, Vol. IT-11, October, 1965.

12. Lee, R.C.K., Optimal Estimation, Identification and Control, MIT Press, 1964.

13. Parzen, E., Modern Probability Theory and Its Applications, John Wiley and Sons, Inc., New York, 1960, (pp. 414-417).

14. Choi, K., "Estimators for the Parameters of a Finite Mixture of Distributions", Tech. Rept. No. 18, University of Missouri, Columbia, Missouri, December, 1966.

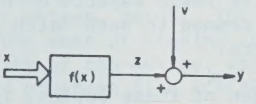


FIGURE 1. Block Diagram of Memoryless System

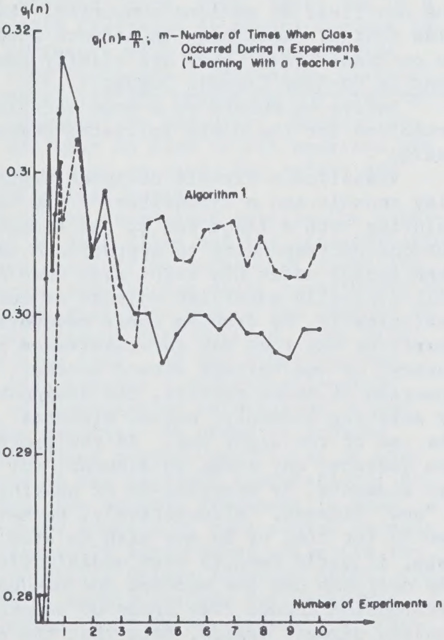


Figure 3. $g_1(n)$ vs. n

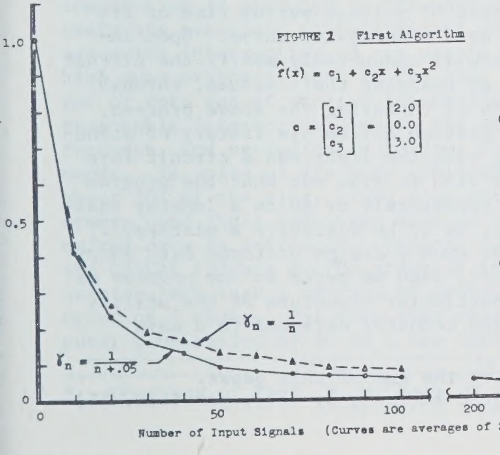


FIGURE 2 First Algorithm

$$\hat{f}(x) = c_1 + c_2x + c_3x^2$$

$$c = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 2.0 \\ 0.0 \\ 3.0 \end{bmatrix}$$

TITLE: ON-LINE CIRCUIT DESIGN; AN INTRODUCTION

AUTHOR: MICHAEL L. DERTOUZOS
Department of Electrical Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts

ABSTRACT

This paper identifies and introduces in simple terms a broad view of on-line circuit design, which reflects the interrelation and relevance of several diverse areas such as network theory, linguistics, numerical analysis, programming systems and on-line computation. Discussion is presented in terms of the structure of on-line utilities, requirements imposed by the interactional nature of on-line circuit design, program structures (data structures) and network-analysis considerations.

INTRODUCTION

General

The recent, and by now well established, development of on-line digital- computer utilities [1,2]* and the more recent emergence of on-line circuit-design programs [3,4,5] have created a new design framework with promising educational and industrial repercussions [6]. The purpose of this paper is to identify and present in simple terms a broad view of what are starting to become the elements of on-line circuit design. This breadth is necessary, since in order to efficiently understand and develop on-line circuit design systems, several fields must be transcended--network theory, "linguistics", numerical analysis, programming-systems, and the new field of on-line computing. Several of the basic aspects of on-line design, especially in network analysis, are common to both batch and to on-line; hence, they are briefly summarized here.

What is On-Line Circuit Design

Before we embark on a more detailed discussion of these topics, let us establish for the newly initiated reader what we mean by "on-line circuit design".

Visualize a circuit designer seated in front of a CRT graphical display console and a typewriter.** He can compose a circuit schematic by pointing with a light pen at the screen and by pressing push (or light) buttons corresponding to appropriate network elements; these elements will then appear under his light pen. Once a circuit is thus composed, the user will typically establish voltage or current excitations, by defining them analytically, by drawing their waveforms, or by naming some standard wave shape; he may then ask for results--a plot of voltage versus time or frequency, or one voltage versus another, or a sensitivity curve. Upon inspection of these results, the designer will undoubtedly modify the circuit by deleting elements, adding elements, or changing their values, through the use of the light pen. In the course of repeating the above process, the designer may model an element, not available from his library of standard elements, by wrapping-up or nesting with the light pen a circuit into a "new" element. Alternatively, he may wish to find out what the program can do for him, or he may wish to stop prematurely or guide a lengthy analysis, if early results seem undesirable, or if he discovers a mistake. *If the designer and the machine can conduct such a design dialogue with short interaction delays (the order of seconds), then we refer to the process as on-line circuit design.* Note that the particular structure of the utility (one-time-shared computer, or a satellite computer attached to a main computer, etc.) is irrelevant to this "definition".

---

* Bracked numbers refer to references at the end of this paper.
** On-line circuit design can be conducted entirely by teletypewriter. [5]

In principle, the same design process can take place in *batch circuit design,* with longer interaction delays, since now it entails the coding of cards and their submittal to and retrieval from the batch computer utility. Realistically, however, there are several fundamental differences [6,7] between these two modes of design, as summarized in the conclusions.

THE ON-LINE UTILITY

The first prerequisite for on-line circuit design is an on-line digital-computer utility. Such a utility consists of (i) computer hardware and (ii) supervisory software. The hardware structure consists of processor(s), memory(ies) and input-output equipment. Components of the structure vary from one utility to the other, but all utilities strive to present a fairly uniform user-visible system. We shall discuss the Project MAC [1] utility which was used for two existing on-line circuit-design programs [3,5] and we shall give brief mention to two other utility approaches.

The Project MAC Utility

As shown in Fig. 1* , the hardware of the MAC system is organized for time-sharing of a modified IBM 7094 processor by up to 30 typewriter users, two graphical displays [7], and background equipment (magnetic tapes, card printers and readers). Here, or interest to on-line circuit design, are the teletypewriters and graphical displays, the use of which has already been described [3,5]. The hardware installation of Fig. 1 is under the control of the software supervisor, two main functions of which are (i) equipment management, and (ii) resource allocation.

Under the numerous functions of equipment management, we distinguish the coordination of various levels of storage--drum, disk, tape, and to a lesser extent, cards. The reason for this diversity in memory devices is that the cost of storing information decreases with increasing storage capacity of the device, while access time increases. Typically, circuit-design programs like CIRCAL [5] and AEDNET [3] are stored on the disk in translated machine code, ready to be "resumed" by the user. Source-language descriptions of these programs are kept in disk if the programs are undergoing revision, but are more often stored on cards for safety and storage economy, until needed for modification purposes.

The resource allocation function of the supervisor entails the scheduling of user requests, the management of user files and the transportation of information. The scheduling of user requests is perhaps of immediate interest, since it determines the average interaction delay experienced by the user. At one extreme, if a task requiring more than a few seconds of computation is executed in its entirety without interruption, then other users are penalized with increased waiting time. At the other extreme, excessive interruption of one user's task creates a disproportionately high swap overhead, that is the cost of transporting information in and out of core and of storing intermediate results. In order to achieve reasonable interaction delays, the supervisor assigns priorities to user requests, and periodically revises them on the basis of computational demands. In spite of its high efficiency, the central processor is idle about 50% of the time, since many programs operate on data stored in mass memory, and their execution must be frequently interrupted while the required data is retrieved. Typical interaction delays in using AEDNET [3] and CIRCAL [5] are of the order of ten to twenty seconds for tasks requiring less than 5 seconds of computation, such as for networks of the order of 5 nodes. For more time consuming analyses, total user-to-computer time ratios of 30 to 1 are typical.

---

* Courtesy of Prof. R. M. Fano, M.I.T.

Other Utilities

Another way of organizing an on-line service, GRAPHIC-1 [8], has been used to implement the on-line circuit analysis system OLCA [4]. It consists of a central processor with a satellite computer. As shown in Fig. 2**, the designer communicates with a "local", PDP5, computer which directs all problem-defining activities at the CRT display. When ready for circuit analysis, the central computer, an IBM 7094, is called into service, and analysis results are returned to the local computer for display. This basic idea--of using a satellite (smaller) computer for tasks demanding little computation, such as circuit composition, modification and display, and resorting to the more powerful machine for the "demanding" analysis task--leads to several advantages, most important of which is a shorter average interaction delay. It also leads to a different utility picture, where service facilities available to a user may not be as diverse. The economic implications of the satellite--utility approach, compared to the single high-power utility, are currently under active debate; it seems that realistic answers can only be given through time and usage.

Another utility approach, which has been used in computer-aided circuit design, involves the perturbation of existing successful batch-design programs for on-line use by a single designer, who alone occupies an entire computer. The cost of such an approach is staggering, since user time, including thinking and resting, equals computer time.

Several other ways for structuring on-line utilities have been reported; we refer the interested reader to the literature [11,38,39] although a number of current approaches are still not documented at this writing. Conspicuous among forthcoming utilities are the MULTICS [9] and the IBM 360/67 [10] systems, both of which employ some form of parallel processing. The former, which succeeds MAC, deviates significantly from the early (MAC) concept of one central time-shared processor. It is distinguished by interjection of a huge "buffer" core memory between the users and several processors, and by the use of several novel information-structuring features, such as segmentation [12].

The foregoing on-line utilities are summarized in Fig. 3 as (i) the single time-shared processor, (ii) the satellite computer, (iii) the batch-converted-to on line- approach and (iv) the memory-centered multi-processor utility. It is not clear at this point which of these utility structures will prove economically efficient. Regardless, however, of what utility configuration will ultimately persist, it is clear [1] that it will strive to present the illusion (or reality) of a wealth of computing services which are "instantly" under the user's command, as if the latter had the constant attention of a huge computer. This apparent trend in on-line utilities transcends into the objective, of on-line circuit-design programs, as evidenced by our "definition" of on-line circuit design of Section 1B, above.

INTERACTIONAL REQUIREMENTS

The on-line utility, and more specifically, the on-line circuit-design environment, create certain requirements, which must be met by effective programs, and which are generalized into five essential classes, as follows:

1. Editing Requirement This requirement entails provisions for inputting and modifying already inputted information. This information may be (i) a network description, (ii) a new element description, (iii) analytical, graphical or tabular representation of a function, which may be an excitation waveform or a nonlinear-element constraint, (iv) new "commands" to be added to existing system commands, and (v) informational or clarifying statements concerning user augmentation of the program through the

---

** Courtesy of Dr. Hing So, Bell Telephone Laboratories definitional ability, described below.

Of the foregoing, most common is the editing of networks. This is minimally accomplished through two operators, one for adding and one for deleting network components. The other editing requirements can usually be satisfied by an operator which permits the input and modification of character strings, without regard to their meaning; such meaning is decided interpretively by other operators, which are later called to operate upon the strings.

Editing is usually accomplished through either a teletypewriter, a graphical console, push buttons, light buttons or combinations of these devices.

2. Output Requirement This requirement entails provisions for outputting of inputted, or computed information. This information may be (i) a network representation, (ii) an element description, (iii) a function representation or a function graph or (iv) operating instructions.

The most common output requirement is the display of analysis results in the form of function graphs--such as the voltage-versus-time waveforms--and the display of edited network schematics. Graphical displays are ideal for satisfaction of this requirement, because of their high plotting speed and the fact that circuit designers are accustomed to oscilloscope waveforms. Typewriters, on the other hand, offer economical and effective plotting media, at the penalty of slower speed.

3. Definitional Requirement Existence of a definitional ability permits a program (and its users) to grow, develop and adapt to prevailing demands. Some typical definitional tasks include (i) definition of "new" nonlinear resistors, inductors and capacitors, i.e. definition of two-variable constraints, and (ii) creation of new "elements" by nesting or encapsulating into one element a network (which may recursively consist of other nests...); (iii) a less typical definitional task is the on-line creation by the user [9] of new commands or operators which may then edit networks on the basis of output results--this leads to a generalized user-controlled optimization capability. Alternatively, such "new" commands may be introduced to display any specified function of element branch, node, voltage, current, time, frequency, etc.

4. Informational Requirement As we discussed in 3 above, an on-line circuit design program, especially a growing one, must be capable of providing the designer with guiding information, at his request. Such information may involve (i) identification of defined terms, such as "what are all the defined resistors" or "how was transistor j defined", (ii) instructions for the use of the system--from brief reminders to lists of possible alternatives and detailed excerpts from the user's manual, (available on-line); and (iii) navigational "fixes", such as "where am I?" and "what have I done so far?"

5. Diagnostic Requirement As programming systems become increasingly complex, progressively fewer mistakes are sufficient to confuse the user or the program. The diagnostic requirement necessitates examination by the computer of user actions, in order to keep such mistakes to a minimum. The classes of errors that a diagnostic ability can detect are usually (i) user errors of form or format, (ii) user violation of problem-oriented "laws" (such as two ideal voltage sources in parallel, a Lipschitz condition which is not met, two totally resistive tunnel diodes in series, and so forth), and (iii) program troubles such as numerical instabilities and the inability of the program to supply requested information.

PROGRAM STRUCTURES

General Program Organization

Programs for on-line circuit design are viewed by their users as a collection of service classes which correspond to the requirement classes of the preceeding Section. Thus, as shown in Fig. 4, at any given time the designer is either using one of these service classes or is evaluating results and deciding about which class to use next. The user is also aware of the existence of a program supervisor which allows selection of appropriate commands within the desired service class, and which "takes care" of all other internal program matters. This "view" is simple, as it should be, for the designer's main task is to design circuits rather than to worry about programs.

The internal program structure, on the other hand, is generally not so simple. A generalization of such a structure is shown in Fig. 5. Here, diamond-shaped blocks correspond to operators (programs, procedures) and rectangular blocks correspond to stored information. The machine part of the on-line circuit-design dialogue is conducted by selective action of these operators upon stored information, resulting in the generation of "new" or computed information.

The operators (more precisely, the operator classes) of Fig. 5 correspond to the five service classes of Fig. 4. Each of these operators generally consists of several other subordinate operators, as shown in Fig. 5 for the output operator. Data storage is divided into two classes: (i) network (and result) data-structures and (ii) "files" representing standard and defined items. Data structures refer to specific "networks" of elementary data units, so arranged, that storage and subsequent processing of that data is efficiently conducted--they are explained in greater detail in the following Section. The remaining data is generally stored as (i) source-language character strings inputted by the user (e.g. instructions), (ii) intermediate data-structures (e.g. for defined functions and nests) or (iii) machine-code "precompiled" data (e.g. standard elements and standard functions).

Standard stored items reflect the foresight of the program designer and the intersection of user needs. They form the "core" of the program on which specialized extensions are later constructed. This data is therefore in precompiled machine-code form. For example, a standard function such as *sine* is stored as a machine-code routine which, given x, generates sinx. Defined items, on the other hand, are appended by various users to enrich and specialize the program. These definitions are usually made in the user's source language (graphics, special typewriter formats, etc.). They generally necessitate interpretors or evaluators, i.e. operators which need not be first translated and then executed, but which are executed on-line following their definition [5].

Information flow in Fig. 5 is as follows: The user creates and edits files such as network descriptions, nests, and functions. These files are either stored intact or, after being processed, by the definitional operator. Note here the recursive flow, that is standard and defined functions and networks may be recursively used to create defined functions and networks (nests). At this first stage, networks are usually stored as the character strings inputted by the user. The output operator converts such network descriptions into a data structure--a form of convenient "plug"--from where analysis operators generate analysis results. During the analysis process, i.e. during execution, interpretors are brought into action to evaluate functions. Finally, the analysis results are made available to the user, and less frequently, to the program, for examination and editing of the network.

Data Structures

In "older" days, information was stored within computer memory, in contiguous storage registers, usually in the form of one-dimensional arrays. With the introduction of list-structure primitives [13] and the formalism of LISP [14], it became possible to represent and store networks of lists, where the lists themselves may be lists of some atomic entities and/or other lists. The need to "structure information in ways more complicated than linear arrays is perhaps best demonstrated by example.

Consider the task of storing a directed graph, such as the one shown in Fig. 6. This graph may represent a network of diodes, but we are not concerned here with its "meaning". One way of storing an n-node directed graph is to reserve an n x n matrix of binary cells, as shown in Fig. 7 and to enter a ONE in cell ij if and only if a directed branch points from node i to node j. Evidently, this representation is one-to-one with the graph, since we can reconstruct the graph from the matrix. As n increases, however, and if the graph is very sparse (dense), a significant amount of memory space, almost proportional to n^2, is wasted with ZERO (ONE) entries.

Consider next a data structure for this graph, based on the existence of pointers, i.e. mechanisms for forming list structures as used in the AED [15] language, or in PL-1 [40]. A pointer links unidirectionally two data structures A and B as follows: In some address allocated to A we store the "head" or leading address of B; we then say that A points to B. In the AED language, as in any list-processing language, convenient mechanisms exist for constructing and taking apart these pointers. Returning to the graph, let us agree that if from node i there emanate k directed branches to nodes 1,... k, then k+1 storage cells are "borrowed" from "free storage" to represent that node. The first of these cells, the head, contains the node number i; the remaining k cells point to the heads of cell structures representing nodes 1, ..., k. This structure is shown in Fig. 8, where we have used consecutive memory addresses, starting from address 100. If we transform Fig. 8 into Fig. 9 by illustrating the pointers graphically with directed branches, then the resulting data-structure is "identical" to graph D. As the number of nodes increases, the pointer-oriented data structure results in significant memory savings, since its memory requirements grow roughly as the square root of the matrix requirements for typical sparse networks. Besides such a *storage efficiency,* data structures make possible a *processing efficiency,* as will be shown next.

Data structures can be "fashioned" in a way, that a particular processing of the stored information can be made quite easily and often independently of the "structure" and "volume" of such information. As an example, consider the task of solving a quasilinear* resistive network iteratively through a technique [16,19] which perturbs at each iteration the voltage of node k as a function** of the value of that voltage and of the surrounding voltages (i.e. the voltages at the other side of each resistor connected to that node), at the previous iteration step.

A data structure [5,19,20] well suited for this application is shown for the circuit of Fig. 10a in Fig. 10b. Here pointers (heavy lines) starting from each head node go toward a cell marked END; they pass through shaded boxes, one for each resistor connected to that node. From each shaded box, one pointer goes to a cell which represents the corresponding nonlinear resistor (by storing i-v characteristics, polarity, etc.) and another pointer (dotted line) goes to the node at the other side of that resistor. To perform one iteration on the network, the following process takes place: Starting from node 1, a "mouse" follows the heavy pointer until the first shaded box; there it "asks" for the other-node voltage, to which the shaded box points with a dotted arrow. Thus, the voltage across the resistor, corresponding to that shaded box, is obtained. Reference is then made to that resistor's characteristics through the other shaded-box pointer, and the current contributed by that resistor is thus obtained. The "mouse" then continues to the next shaded box where it repeats the above process, obtaining the current contributed by the corresponding resistor. That current is added to the previous current and the "mouse" continues until the END, thus obtaining the total current incident on that node. The entire foregoing process is repeated for each of the remaining nodes, and then the node voltages are perturbed as a function of the computed currents.

---

* All nonlinear resistors have continuous monotonically increasing current-voltage characteristics with bounded non-zero slopes

** More precisely, a function of the total current incident on that node; that current is not zero until the algorithm has converged on a solution.

Suppose now that in the course of on-line design, a resistor is removed from the network. The corresponding operation in the data structure is to delete the corresponding two shaded boxes (on each side of the resistor) and the cell storing the properties of that resistor. If a new resistor is to be added to the network, then the inverse operations are performed. *In either case, regardless of topology and network size, the "mouse" (or algorithm) continues to do exactly the same relatively easy and viciously repetetive task, and so does the editing operator which inserts or removes elements.*

There are obviously many data structures possible for the same network. For example, pointers passing through elements that possess common properties such as all capacitors, resistors, and inductors, have already been used in an on-line circuit design program [3]. Of all the possible data structures, however, we conclude our discussion with an example which illustrates the representation of recursive constructs.

Suppose that the network of Fig. 10a is "nested" into the element NEST1 of Fig. 11a, which is, in turn, used in the network CIRCUIT1 of Fig. 11b. The data structure for CIRCUIT1 is constructed as before, with the exception that each terminal of NEST1 will be represented by a shaded circle (a special kind of storage cell) rather than by a shaded box. Part of the resulting data structure is shown in Fig. 11c, from where it is seen that the purpose of the shaded circle is to provide an exit and a return for the "mouse", which, starting from node 19, continues to "see" the same form of data structure as in Fig. 10b. Thus, the iteration algorithm remains unchanged, except for being able to distinguish the shaded circles. Evidently, the above recursive aspects of "nesting" can be carried to any depth, *without increasing the complexity of the solution algorithms or that of the mechanisms for setting up data structures, or for editing, and nesting networks.* This conclusion is not surprising, since list-processing languages, such as LISP, were intended [21] on the first place as formalizations of recursive-function representations.

In conclusion, data structures make possible (i) efficient use of storage, (ii) efficient application of algorithms, (iii) the use of operators which are independent of the size and structure of stored information, and (iv) efficient representation and processing of recursive constructs.

ANALYSIS

Network analysis in an on-line environment is, generally, similar to batch network analysis. There are, however, several differences caused by the interactional nature of on-line design, such as (i) premature termination of analysis, by the designer, as soon as program troubles or bad design "moves" are identified, (ii) changes in the course of analysis, designated by the designer, as results are revealed; such as, for example, the concentration of more computing power on some portion of the network or of time. This adaptive way of analysis can be particularly effective in the case of long solution times, because it provides the designer with the ability to "guide" analysis, rather than to preset it and then possibly suffer the "consequences" of a long delay for unwanted results.

Besides this interactive potential, network analysis for on-line circuit design, as in the case of batch analysis, is by no means governed by a universal approach. Techniques used in the several hundreds of "programs" currently existing and surveyed [22] in the literature, may be classified in several "orthogonal" ways. They may be classified* for example according to the *type of networks* that they treat, such as (i) linear resistive [23], (ii) linear ladder-like [24], (iii) linear (or "slightly" nonlinear) RLC with (without) dependent sources [25,26], (iv) quasilinear resistive [19], or (v) nonlinear RLC with constraints on topology and characteristics [3,5]. Analysis techniques may also be classified according to their *dynamics,* i.e. (i) time-domain [5], (transient) (ii) frequency domain [27] (iii) static (D.C.); or according to *special features* such as (i) sensitivity and (ii) optimization capabilities. This diversity is further aggravated, since for a given network class and dynamics, there remains the *formulation* task, i.e. the extraction from the network of characterizing equations. Here, formulations may be classified according to dynamics, i.e. (i) nodal [23], or mesh [28], and (ii) tree link [29], and according to dynamics, i.e. (i) state-variables [30,31,32], and (ii) system (transfer) functions [33]. As if the combinatorial problem were not already big enough, tearing schemes may be introduced, in order to partition a network into subnetworks, which are first treated by some combination of the above classifications, and their solutions are then joined together. *Tearing* a network may induce the following subnetwork classes: (i) linear-nonlinear [5,25], (ii) fixed and variable [27] (from analysis to analysis), (iii) memory and memoryless [5], (iv) three (R,L,C) one-element subnetworks [3], and (v) several smaller subnetworks [34]. Even after all of the above have been established for some analysis technique, there are possible several classes of numerical methods for solving the formulated equations. The types of equations predominant in network analysis are (i) systems of (algebraic) equations and (ii) ordinary vector differential equations (arising from the state variable approach). The former may be solved by (i) matrix inversion (linearized if nonlinear) or by iteration, and the latter by the so-called intergration techniques such as Runge Kutta, Adams-Bashforth, predictor-corrector [35], or Certaine's method [36].

---

* References indicate one or two relevant papers and are not intended to be complete.

CONCLUSIONS

The on-line approach to circuit design, besides affording savings in design time, is distinguished by the more general properties of *adaptive* processes (or experiments) as contrasted to *preset* processes. An adaptive process is an experiment, where a sequence of inputs generates a sequence of outputs, and at any given time, the next input to the experiment may depend on observation of all the previous outputs. A preset experiment, on the other hand, involves a string of inputs generated without knowledge of the outputs. The interested reader will find that adaptive experiments, in the theory of finite-state machines [37], often lead to results which are either not possible or more difficult with preset experiments.

In circuit design, this "adaptive" process makes possible: a progressive accumulation of accurate and diverse knowledge about circuit performance, before circuit fabrication. This is accomplished through modifying the network, observing results, and defining necessary extensions of the system. This interactive process is further enriched with the convenience of informational and diagnostic services, and the ability to "guide" analysis.

In order to understand and construct effective on-line circuit-design programs, several otherwise diverse issues must be considered in relation to each other. They are: (i) the software and hardware structure of on-line utilities, (ii) the requirements imposed upon the program by the circuit designer and by the interactional nature of on-line design, (iii) the organization of operators and data (data structures) for efficient circuit-analysis programs and (iv) the mathematical foundations of network analysis and numerical analysis. It was the purpose of this paper to identify in an introductory manner, the above issues as they emerge from the early development of on-line circuit design. In spite of this early development, however, it is already becoming apparent that implications of on-line circuit design are quite far-reaching and will undoubtedly occupy an important place in the design and educational tools of electrical engineers.

ACKNOWLEDGMENT

I wish to thank my graduate students at MIT who are providing continuous inspiration and valuable contributions, and who have helped make this research interesting.

This work was supported in part by the National Aeronautics and Space Administration under Research Grant NsG-498 (Part) and in part by Project MAC, an MIT research program sponsored by the Advanced Research Projects Agency, Department of Defense, under Office of Naval Research Contract Nonr-4102(01). Reproduction in whole or in part is permitted for any purpose of the United States Government.

REFERENCES

1. P. Henrici, Elements of Numerical Analysis. New York: Wiley, 1964.
2. F. J. Corbato, M. M. Daggett, and R. C. Daley, "An Experimental Time-Sharing System," Proc. SJCC, vol. 21, pp. 335-344, 1962.
3. J. Katzenelson, "AEDNET: A Simulator for Nonlinear Networks," Proc. IEEE, vol. 54, p.p. 1536-1552, November 1966.
4. Hing C. So, "OLCA: An on-Line Circuit Analysis System," 1967 IEEE International Convention Record, also Proceedings of the Computer-Aided Circuit Design Seminar by N.A.S.A. E.R.C. Cambridge, Mass., April 11-12, 1967.
5. Michael L. Dertouzos, "CIRCAL" On-Line Circuit Design," Proc. IEEE, vol. 55, no. 5, pp. 637-654, May 1967.
6. Michael L. Dertouzos, "On-Line Circuit Design: Some Capabilities and Limitations." Position Paper, Proc. of the IEEE, November 1967.
7. R. H. Stotz, "Man-Machine Console Facilities for the Computer-Aided Design System," Proc. SJCC, May 1963.
8. W. H. Ninke, "Graphic 1--A Remote Graphical Display Console System," Proc. FJCC, Spartan Books, Inc., Washington D.C., vol 27, p.p. 834 846,1965
9. F.J. Corbato and V. A. Vyssotsky, "Introduction and Overview of the Multics System," Proc. FJCC, November 1965.
10. IBM System 360: Time Sharing System; Concepts and Facilities, IBM Form No. C28-2003-0.
11. A. L. Samuel, "Time-Sharing on a Computer," New Scientist, June 1965 also Technical Report MAC-TR-17, March 1965, Project MAC, MIT, Cambridge, Mass.
12. J. B. Dennis, "Segmentation and the Design of Multiprogrammed Computer Systems," IEEE International Convention Record, Institute of Electrical and Electronic Engineers, New York, Part 3, p.p. 214-225,1965.
13. Newell, Allen, Shaw, and Simon, "The Logic Theory Machine," IRE Trans. on Information Theory, IT-2, No. 3, p.p. 61-79,1956.
14. John McCarthy, et. al., The LISP 1.5 Programmer's Manual, MIT Press, Cambridge, Mass.
15. D. T. Ross and J. E. Rodriguez, "Theoretical Foundations for the Computer-Aided Design System," Proc. SJCC, May 1963.
16. G. Birkhoff and S. B. Diaz, "Nonlinear Network Problems," Quart. Applied Math., vol. 13, p.p. 431-443, January 1956.
17. R. V. Southwell, "Stress Calculation in Frameworks by the Method of Systematic Relaxation of Constraints, I and II," Proc. Roy. Soc. (London), ser. A, vol. 164,1938.
18. J. Katzenelson and L. Seitelman, "An Iterative Method for Solution of Nonlinear Resistor Networks," in Proc. 3rd Allerton Conf. on Circuit and System Theory, Urbana, Ill.: University of Illinois Press, 1965.
19. M. L. Dertouzos and C. W. Therrien, "CIRCAL: On-Line Analysis of Electronic Networks," M.I.T. Electronics Systems Lab., Cambridge, Mass., Rept. ESL-248, October 1965.

20. C. W. Therrien, "A Digital Computer Simulation of Electrical Networks," S. M. thesis, Dept. of Electrical Engineering, M.I.T., Cambridge, Mass., June 1965.
21. Marvin Minsky, Computation; Finite and Infinite Machines, Prentice-Hall, Inc., 1967.
22. F. F. Kuo, "Network Analysis by Digital Computer," *Proc. IEEE,* vol. 54, p.p. 820-829, June 1966.
23. C. W. Rosenthal and H. Simon, "Computers as an Aid in the Development of Transmission Circuits," *1964 Proc. 2nd Annual Allerton Conf. on Circuit and System Theory*, p.p. 337-365.
24. T. R. Bashkow, "Network Analysis," in *Mathematical Methods for Digital Computers*, A. Ralston and H. S. Wilf, Eds. New York: Wiley, 1960, ch. 26.
25. A. F. Malmberg, F. L. Cornwell, and F.N. Hofer, "NET-1 Network Analysis Program," Los Alamos Scientific Lab., Los Alamos, N.M. Rept. LA-3119 August, 1964, 7090/94 version.
26. "1620 Electronic Circuit Analysis Program (ECAP)," *User Manual*, IBM Rept. 1620-EE-02X.
27. H. C. So, "On the Hybrid Description of a Linear n-Port Resulting From the Extraction of Arbitrarily Specified Elements," *IEEE Trans. on Circuit Theory*, vol. CT-12, p.p. 381-387, September 1965.
28. R. A. Manmano, K. D. Pope, and E.P. Schneider, "Computer-Assisted Circuit Analysis," *EDN*, p.p. 132-139, November 1965.
29. H. Pachon, "D. C. Automatic Electric Network Analysis," 1620 User's Group Meeting, Miami, Fla. May 1965.
30. T. R. Bashkow, "The A Matrix, New Network Description," *IRE Trans. on Circuit Theory*, vol. CT-4, p.p. 117-120, September 1957.
31. F. H. Branin, Jr. "DC and Transient Analysis of Networks Using a Digital Computer," *1962 IRE International Conv. Rec*, Part 2, p.p. 236-256.
32. P.R. Bryant, "The Explicit Form of Bashkow's A Matrix," *IRE Trans. on Circuit Theory*, vol CT-9 p.p. 303-306, September, 1962.
33. M. L. Liou, "A Novel Method of Evaluating Transient Response, "*Proc. IEEE*, vol. 54, p.p. 20-23, January 1966.
34. G. Kron, "A Method of Solving Very Large Physical Systems in Easy Stages," *Proc. IRE*, vol. 42, p.p. 680-686, April 1954.
35. P. Henrici, *Elements of Numerical Analysis*. New York: Wiley, 1964.
36. J. Certaine, "The Solution of Ordinary Differential Equations With Large Time Constants," in *Mathematical Methods for Digital Computers*. A. Ralston and H. S. Wilf, Eds. New York: Wiley, 1960, ch. 11.
37. A. Gill, *Introduction to the Theory of Finite-State Machines*. McGraw-Hill, 1962.
38. M. M. Gold and A. C. Stedry, "An Evaluation of Commercial Time Sharing Systems," Carnegie Institute of Technology. Management Sciences Research Report 71, May 1966.
39. E. Burgess, "On-Line Computing Systems," Proceedings of the Symposium sponsored by the University of California at Los Angeles, February 2-4, 1965, American Data Processing Inc. 1965.
40. Harold W. Lawson Jr., "PL/1 List Processing," *Communications of the* ACM, Vol. 10, No. 6, June 1967, p.p. 358-367.

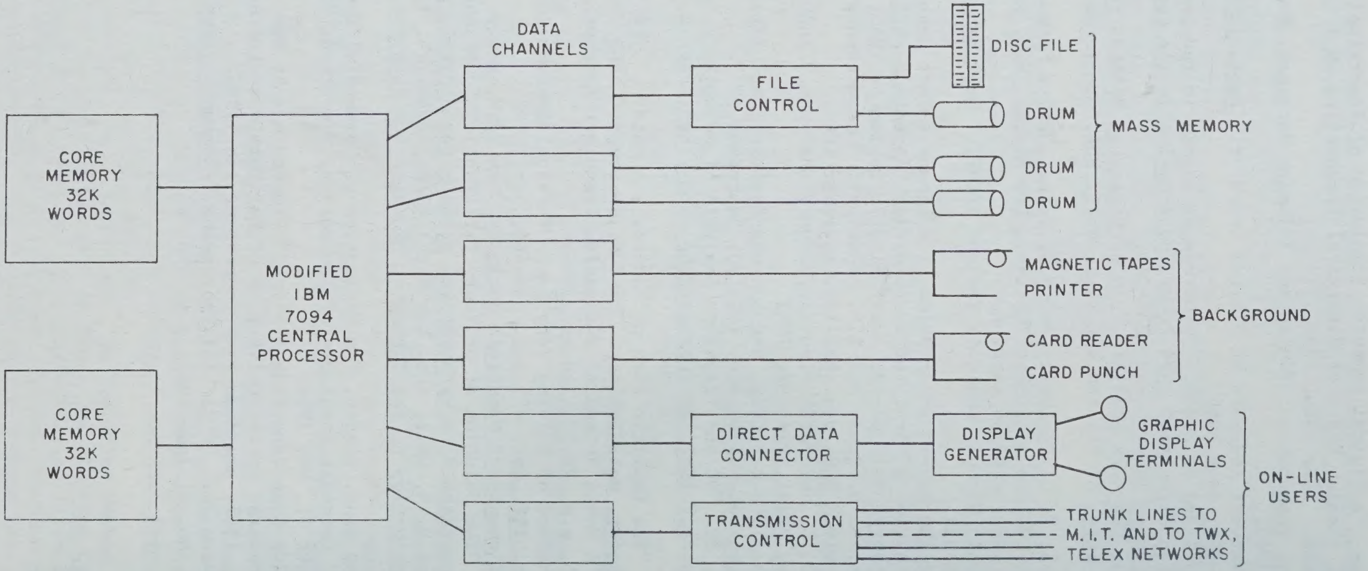

DATA CHANNELS
CORE MEMORY 32K WORDS
CORE MEMORY 32K WORDS
MODIFIED IBM 7094 CENTRAL PROCESSOR
FILE CONTROL
DISC FILE
DRUM
DRUM
DRUM
MASS MEMORY
MAGNETIC TAPES
PRINTER
CARD READER
CARD PUNCH
BACKGROUND
DIRECT DATA CONNECTOR
DISPLAY GENERATOR
GRAPHIC DISPLAY TERMINALS
TRANSMISSION CONTROL
TRUNK LINES TO M.I.T. AND TO TWX, TELEX NETWORKS
ON-LINE USERS

Fig. 1. Hardware Organization of the MAC Utility

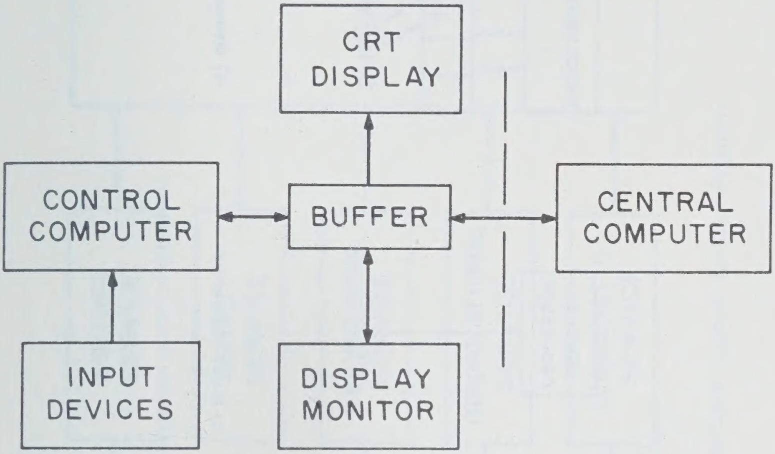
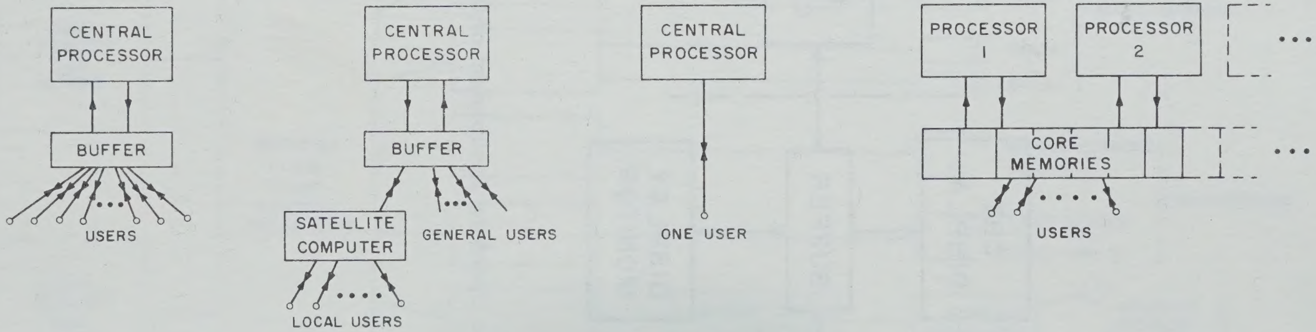

CRT DISPLAY
CONTROL COMPUTER
BUFFER
CENTRAL COMPUTER
INPUT DEVICES
DISPLAY MONITOR

Fig. 2. The Operating Environment of OLCA


CENTRAL PROCESSOR
BUFFER
USERS
CENTRAL PROCESSOR
BUFFER
SATELLITE COMPUTER
GENERAL USERS
LOCAL USERS
CENTRAL PROCESSOR
ONE USER
PROCESSOR 1
PROCESSOR 2
CORE MEMORIES
USERS

a) Single Time-Shared Processor

b) The Satellite-Computer Approach

c) Batch Converted to On-Line

d) Memory-Centered Approach

Fig. 3. Summary of On-Line Utility Configurations

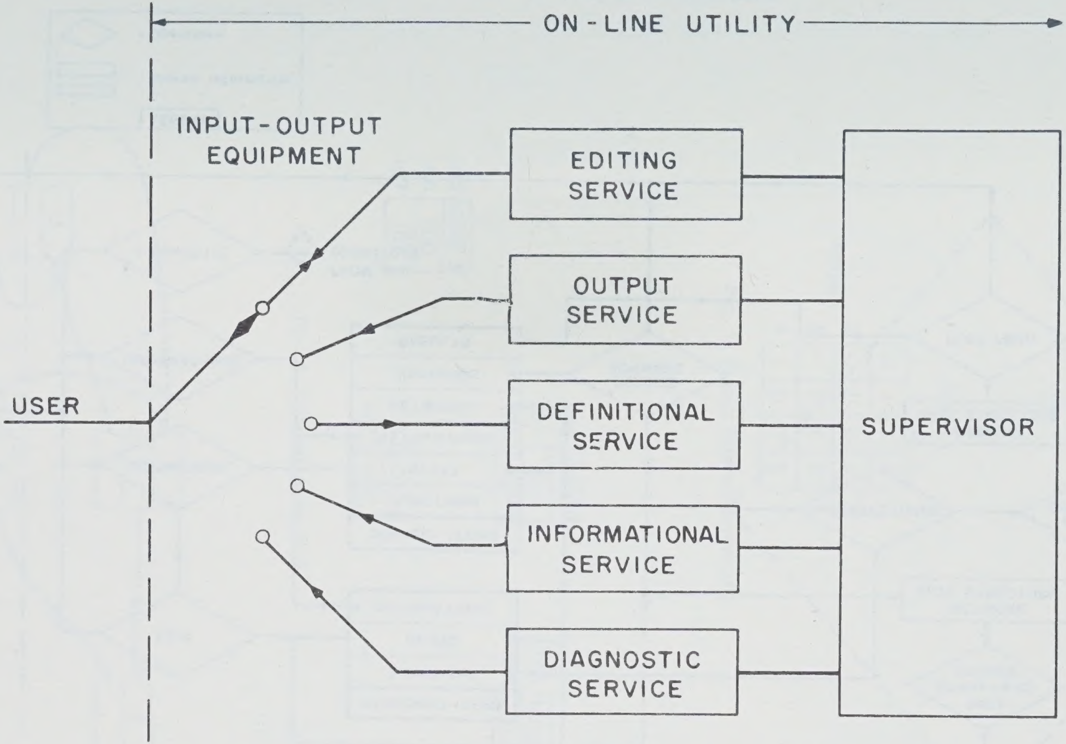

ON-LINE UTILITY
INPUT-OUTPUT EQUIPMENT
USER
EDITING SERVICE
OUTPUT SERVICE
DEFINITIONAL SERVICE
INFORMATIONAL SERVICE
DIAGNOSTIC SERVICE
SUPERVISOR

Fig. 4. User's View of an On-Line Circuit Design Program

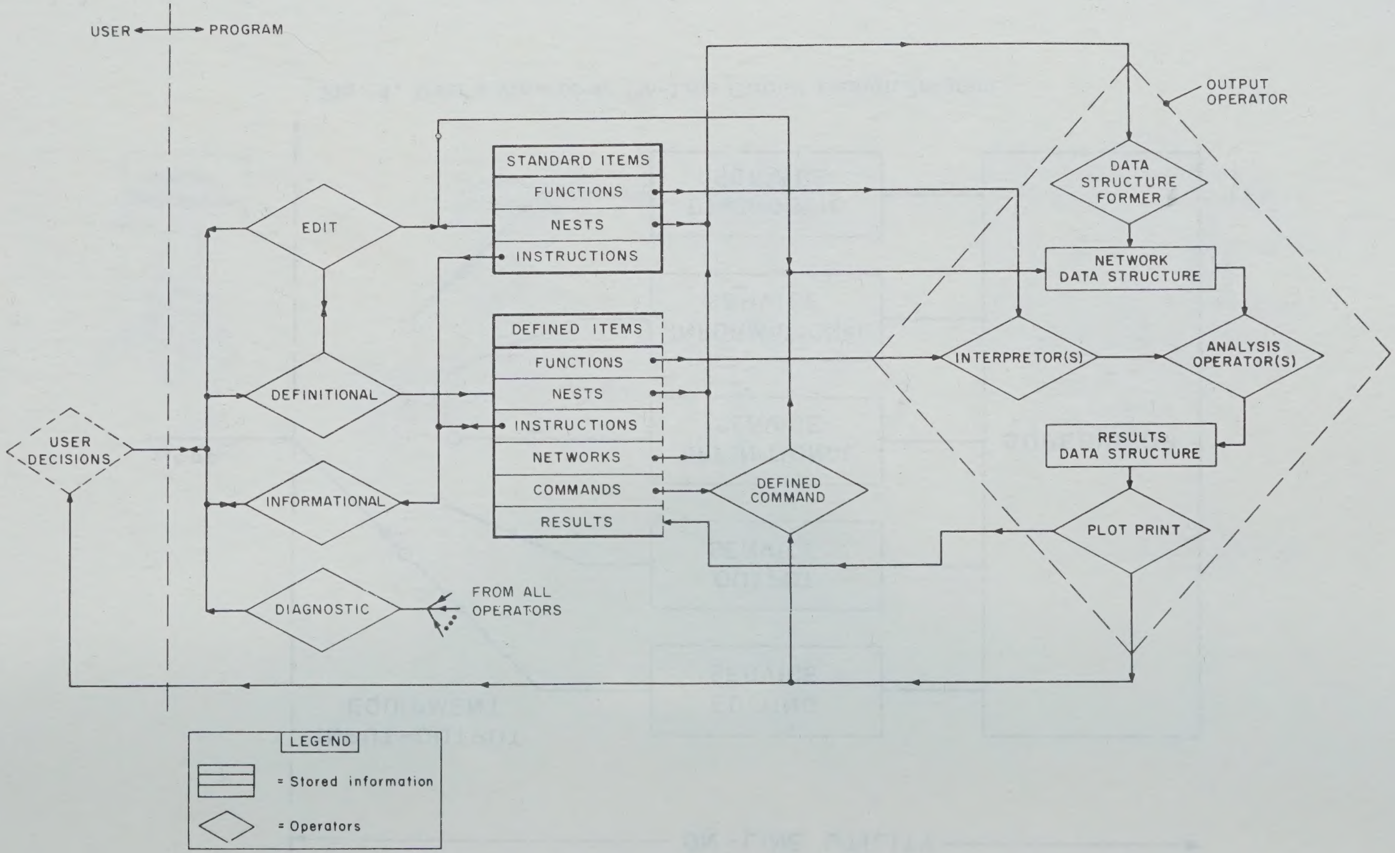

USER
PROGRAM
EDIT
DEFINITIONAL
INFORMATIONAL
DIAGNOSTIC
USER DECISIONS
STANDARD ITEMS
FUNCTIONS
NESTS
INSTRUCTIONS
DEFINED ITEMS
FUNCTIONS
NESTS
INSTRUCTIONS
NETWORKS
COMMANDS
RESULTS
FROM ALL OPERATORS
DEFINED COMMAND
OUTPUT OPERATOR
DATA STRUCTURE FORMER
NETWORK DATA STRUCTURE
INTERPRETOR(S)
ANALYSIS OPERATOR(S)
RESULTS DATA STRUCTURE
PLOT PRINT
LEGEND
= Stored information
= Operators

Fig. 5. Programming-System Structure for On-Line Circuit Design

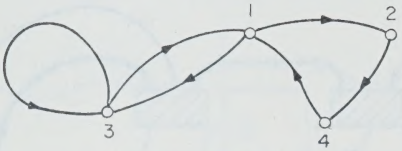


Fig. 6. A Directed Graph, D

$$
\begin{array}{cc|c|c|c|c|}
 & & \multicolumn{4}{c}{\text{NODE } j} \\
 & & 1 & 2 & 3 & 4 \\
\hline
 & 1 & 0 & 1 & 1 & 0 \\
\hline
 & 2 & 0 & 0 & 0 & 1 \\
\hline
\text{NODE } i & 3 & 1 & 0 & 1 & 0 \\
\hline
 & 4 & 1 & 0 & 0 & 0 \\
\hline
\end{array}
$$

Fig. 7. Matrix Representation of D

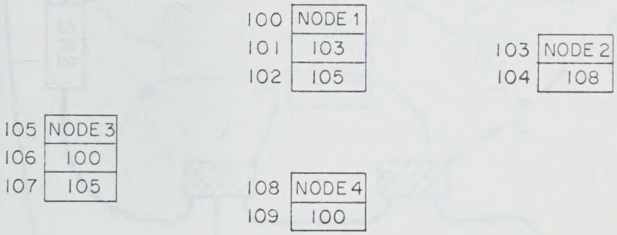

100 NODE 1
101 103
102 105
103 NODE 2
104 108
105 NODE 3
106 100
107 105
108 NODE 4
109 100

Fig. 8. Data Structure of D

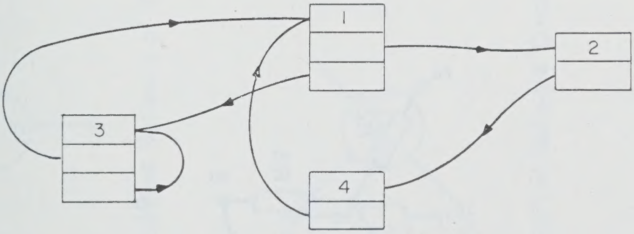


Fig. 9. Redrawn Data Structure of D

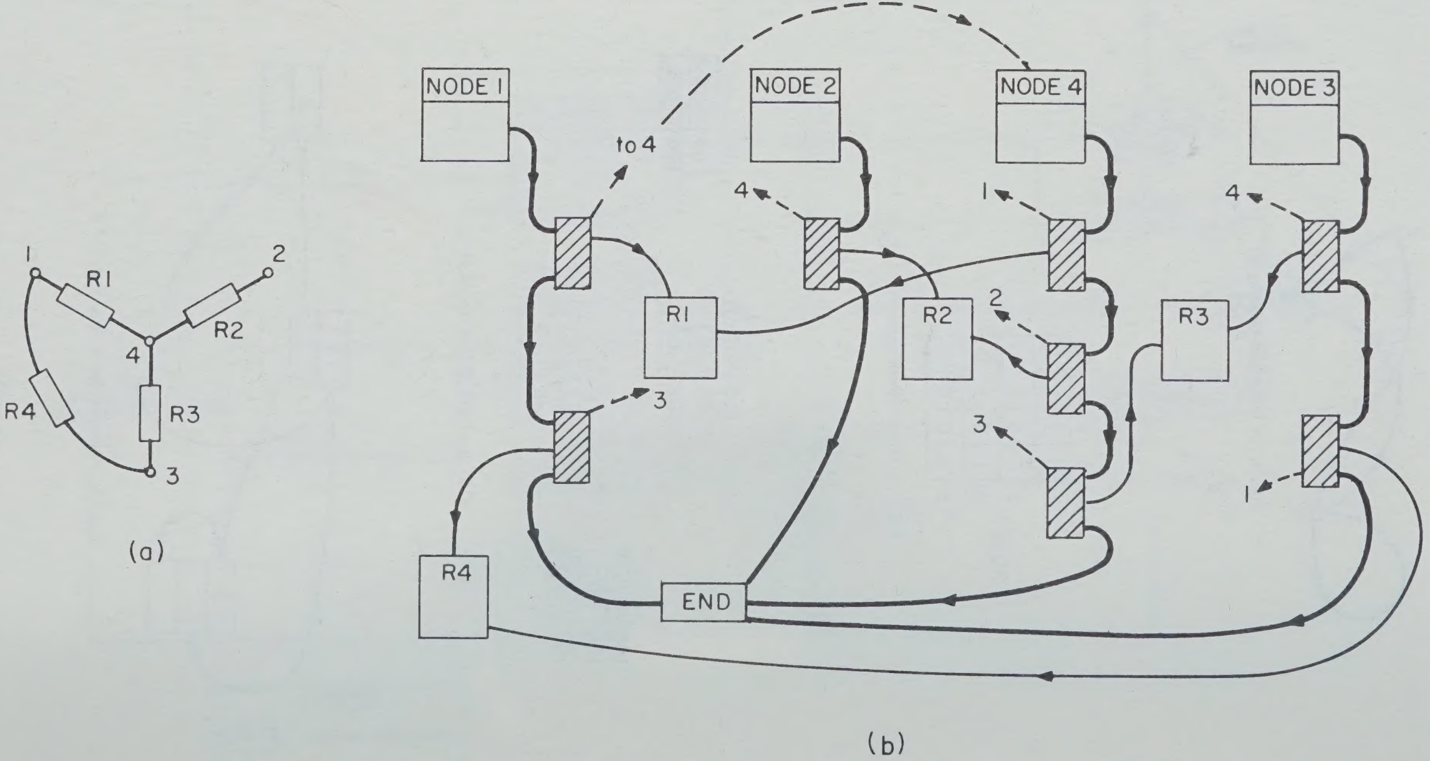
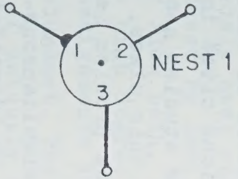
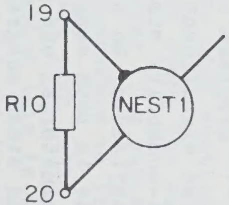

1
R1
2
R2
4
R4
R3
3
(a)
NODE 1
NODE 2
NODE 4
NODE 3
to 4
4
1
4
R1
R2
2
R3
3
3
1
R4
END
(b)

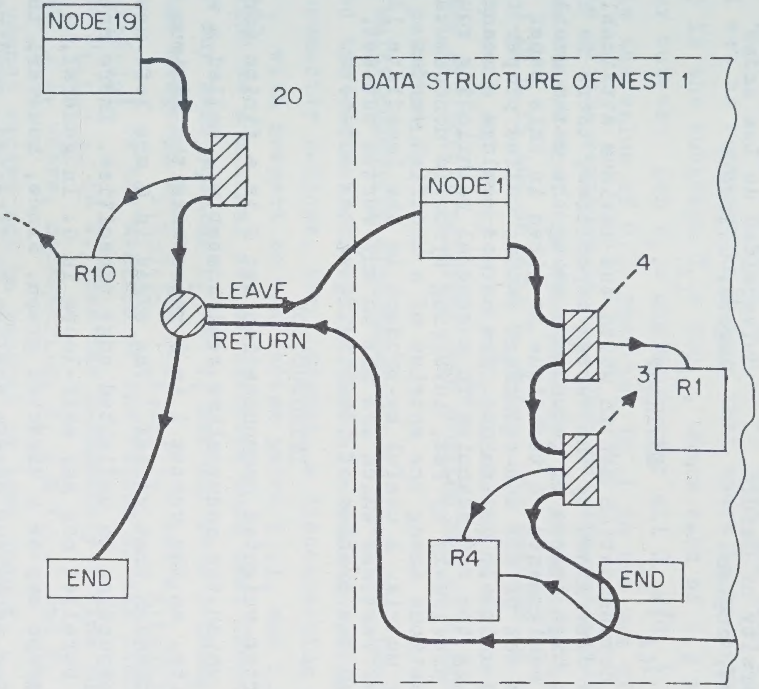
Fig. 10. A Data Structure for Iterative Solution of Resistive Networks



a) The Nested Network of Fig. 10a


19
R10
NEST1
20

b) Part of Network "Circuit 1"


NODE 19
20
R10
LEAVE
RETURN
END
DATA STRUCTURE OF NEST 1
NODE 1
4
3
R1
R4
END

c) Part of the Data Structure of "Circuit 1"

Fig. 11. Use of Data Structures in Nesting

PROPER FLOW PATHS IN VERTEX-WEIGHTED COMMUNICATION NETS

B. R. MYERS
University of Oxford
Oxford, England

E. A. DAVILA
Universidad de Los Andes
Bogota, Colombia

ABSTRACT

In a communication net in which the maximum attainable flows are determined by the vertex capacities, there is in general a high degree of redundancy among the paths available for flow assignments. A technique is given in this paper by which the set of all non-redundant paths, called proper flow paths, is uniquely determined. The method employs a means of eliminating the redundancies in a general pathfinding routine proposed previously by Paz, involving symbolic noncommutative multiplications among the entries of a modified vertex adjacency matrix. A useful by-product is the immediate identification of vertices which are not in any vertex cut-set. This simplifies the problem of identifying vertex cut-sets.

INTRODUCTION

A vertex-weighted communication net G is a finite connected graph in which the nonnegative real number w_i, called a weight or capacity, on the vertex i in G represents the maximum rate of flow through that vertex.[1] The edges in G are left unweighted, corresponding to unlimited edge capacities. There are no edges in parallel nor any self-loops in G. In general, G or some subgraph may be a directed graph. Since, however, the pathfinding technique herein applies to directed, nondirected or mixed graphs, it suffices to assume that G is nondirected. As a model of a physical system G thus represents one in which the station-to-station (vertex-to-vertex) information handling capacities are limited only by the capacities of the switching centers (the vertices) and not by those of the communication channels (the edges).

The state of the art in analysis and synthesis of vertex-weighted nets is represented by the contributions cited in the references.[1,2,3,4,5,6] Progress in single-flow studies has essentially exhausted significant problems in this class. By contrast, the multi-commodity simultaneous-flow problem has proved relatively unyielding to date.[5]

A major obstacle in both single-flow and simultaneous-flow studies is the combinatorial one which arises in identifying cut-sets, paths, and other subgraphs. In single-flow analysis, for instance, suppose $\{C_{ij}\}$ is the set of all vertex cut-sets with respect to the vertices i, j in G. That is, each C_{ij} is a minimal set of vertices in G such that there is no path joining i and j in the subgraph $\overline{C}_{ij} = G - C_{ij}$, where each of i, j is itself a vertex cut-set. Let $V_{ij} = \min|C_{ij}|$ over all $C_{ij} \in \{C_{ij}\}$, where $|C_{ij}|$ is the value of C_{ij}. That is, $|C_{ij}|$ is the sum of the weights of the vertices in C_{ij}. Then the min-cut max-flow theorem states that the flow rate f_{ij} is attainable between i and j in G if and only if $f_{ij} \leq V_{ij}$. The problem is thus to identify each and every cut-set C_{ij} in G. Since cut-sets and paths are explicitly related,[1] the problem can be regarded as one of pathfinding.

In multi-commodity simultaneous-flow analysis, Davila[6] has extended the cut-set concept to that of a disconnecting set. For 2-commodity 2-flows, for instance, a disconnecting set $D_{i_1 i_2 j_1 j_2}$ with respect to the vertex pairs i_1, j_1 and i_2, j_2 in G is defined as a minimal set of vertices such that there is neither any path joining i_1 and j_1 nor any path joining i_2 and j_2 in the subgraph $\overline{D}_{i_1 i_2 j_1 j_2} = G - D_{i_1 i_2 j_1 j_2}$. Let $W_{i_1 i_2 j_1 j_2} = \min|D_{i_1 i_2 j_1 j_2}|$ be the minimum of the values of all disconnecting sets $D_{i_1 i_2 j_1 j_2}$ in G. Then the 2-commodity 2-flow sum $(f_{i_1 j_1} + f_{i_2 j_2})$, where, necessarily, $f_{i_1 j_1} \leq V_{i_1 j_1}$ and $f_{i_2 j_2} \leq V_{i_2 j_2}$, cannot exceed and in general is less than $W_{i_1 i_2 j_1 j_2}$. The problem is to identify each and every disconnecting set in G. Davila has shown how to do this by finding only certain single-flow cut-sets. This, however, does not solve the problem directly but merely reduces it to that which applies in the single-flow case which, again, can be regarded as one of pathfinding.

Though its applicability in solving this problem has not previously been recognized, it happens that another concept introduced by Davila is germane. This is the concept of a proper flow path, defined in the next section. There is in general a high degree of redundancy among the paths available for flow assignments in G. Only proper flow paths are, in fact, non-redundant. The problem is therefore to find a practical means of uniquely identifying all such paths in G. A technique is presented in this paper by which this is achieved. It succeeds through an astute means of eliminating redundancies in a general but otherwise inefficient pathfinding routine proposed recently by Paz.[7] A useful by-product is obtained with the immediate identification of vertices which are not in any vertex cut-set. This simplifies the inescapable problem of finding the vertex cut-sets and disconnecting sets in G.

PROPER FLOW PATH THEORY

The following definition of a proper flow path is a simpler version of Davila's original definition.[6]

Definition 1. A path P_{ij} joining the vertices i,j in G such that no two non-adjacent vertices in P_{ij} are adjacent in G is said to be a proper flow path with respect to i,j in G.

Definition 2. A vertex or an edge which is not in any proper flow path joining the vertices i and j in G is said to be redundant with respect to i,j in G.

Definition 3. The length of a path is the number of edges (and hence one less than the number of vertices) it contains.

While it applies to any graph, the concept of a proper flow path has particular significance in vertex-weighted nets because of the results contained in the lemmas and theorems below.

Lemma 1. If the vertices i,j are adjacent in G, the edge (i,j) is the only proper flow path with respect to i,j in G.

Proof. By Definition 1, any path joining i and j in G, other than the edge (i,j), is not a proper flow path.

Lemma 2. If vertices i,p,q are mutually adjacent in G, then for k any vertex other than i in G the edge (p,q) is redundant with respect to i,k.

Lemma 3. A shortest path joining any two vertices in G is a proper flow path with respect to those vertices.

Proof of Lemma 2. Since i,p,q are mutually adjacent, a proper flow path P_{ik} joining i and k in G can contain at most one of p,q.

Proof of Lemma 3. Let P'_{ij} be a shortest path joining vertices i and j, and suppose that P'_{ij} is not a proper flow path. Then at least one pair of non-adjacent vertices p,q in P'_{ij} are adjacent in G. Let P'_{pq} be the (unique) path joining p and q in P'_{ij} and consider the path P_{ij} in G obtained by replacing the path P'_{pq} in P'_{ij} by the edge (p,q). All of the vertices in P_{ij} are in the path P'_{ij}, and they are fewer in number. Thus, P_{ij} is a shorter path than P'_{ij}. Which contradicts the hypothesis.

Theorems 1, 2, and 3 are next given without proof, since space limitations preclude otherwise. Theorems 2 and 3 are of central significance. The former shows that an attainable flow f_{ij} can always be routed from i to j exclusively through proper flow paths. Thus, nothing is lost by erasing all redundant vertices and edges from G. The residual net G' is generally much simpler than G and therefore more easily analyzed. One consequence of this simplification is evident in Theorem 3, which shows that the set of vertex cut-sets with respect to i,j is the same in both G and G'.

Theorem 1. The set of all proper flow paths with respect to the vertices i,j in G is unique.

Theorem 2. It is sufficient to route any attainable flow f_{ij} between the vertices i,j in G exclusively through proper flow paths with respect to i,j in G.

Theorem 3. A vertex k in G is not in any vertex cut-set C_{ij} which separates i and j in G if and only if k is redundant with respect to i,j in G.

Finally, a theorem which relates to the generation of the set of proper flow paths and which also facilitates achieving the same end when a geometric drawing (topological graph) of G is possible. The following notation is used for a union of path sets. Suppose $\{P_{i_r j_s}\}$ is the set of all paths joining vertices i_r and j_s in G, and that i_r, j_s range over $i_1, i_2, \ldots, i_p$ and $j_1, j_2, \ldots, j_q$, respectively. Then the union of all sets $\{P_{i_r j_s}\}$ over all i_r and j_s is denoted by $\bigcup_{r,s} \{P_{i_r j_s}\}$.

Theorem 4. If the vertices $i_1, i_2, \ldots, i_p$, and only these vertices, are adjacent to the vertex i and the vertices $j_1, j_2, \ldots, j_q$, and only these vertices, are adjacent to the vertex j in G, then the set $\{P_{ij}\}$ of all proper flow paths joining i and j in G is given by

$$\{P_{ij}\} = \bigcup_{\substack{1 \le r \le p \\ 1 \le s \le q}} \left\{(i, i_r) \cup \{P_{i_r j_s}\} \cup (j_s, j)\right\}$$

where (i, i_r) and (j_s, j) are the edges joining i, i_r and j_s, j, respectively.

Proof. By Lemma 2 it may be assumed without loss in generality that neither any two of $i_1, i_2, \ldots, i_p$ nor of $j_1, j_2, \ldots, j_q$ are adjacent in G. Let i_r be any one of $i_1, i_2, \ldots, i_p$ and j_s be any one of $j_1, j_2, \ldots, j_q$. Assume that there exists at least one such pair i_r, j_s such that $i_r \neq j_s$ and such that there is at least one proper flow path $P_{i_r j_s}$ from i_r to j_s in G, this path not containing either i or j; otherwise, there is nothing to prove. Since i,j are not adjacent in G then i,j are adjacent to and only to the vertices i_r, j_s, respectively, which are in $P_{i_r j_s}$. Thus, $(i, i_r) \cup P_{i_r j_s} \cup (j_s, j)$ is a proper flow path joining i and j in G, and the theorem follows.

EXAMPLE NET

In illustration of the application of the previous lemmas and theorems, consider the net G of Figure 1(a), with the vertices labelled a,b,...,k. The single flow f_{ab} between a and b is assumed to be the only flow of interest, for the moment. First, by Lemma 2 the edges (c,d), (i,j), and (j,k) are redundant so far as f_{ab} is concerned and may therefore be erased from G. Next, by Theorem 4 the set of all proper flow paths $\{P_{ab}\}$ may be found by simply finding the sets $\{P_{ci}\}, \{P_{cj}\}, \{P_{ck}\}, \{P_{di}\}, \{P_{dj}\}, \{P_{dk}\}$, each of which can be obtained by inspection in this particular example. The unions $\bigcup_{r=i,j,k} \{P_{cr}\}$, $\bigcup_{s=i,j,k} \{P_{ds}\}$ are shown as subgraphs G_1, G_2, respectively, in Figure 1(b). Whence the subgraph G_3 of G shown in Figure 1(c), which contains all of and only the proper flow paths in $\{P_{ab}\}$. It is apparent from G_3 that none of the vertices in G are redundant with respect to a,b. Notice, however, that edge (e,f) is redundant, as could have been determined earlier by applying Lemma 2 to c,e,f. Also, edge (f,g) is redundant with respect to d,b but not a,b.

By Theorem 3, the set of vertex cut-sets $\{C_{ab}\}$ with respect to a,b is the same in both G and G_3. Since G_3 is a simpler net than G, they are much more easily obtained from the former. Better yet, they can still more easily be obtained from G_1 and G_2. For if C_{cb}, C_{db} are respectively vertex cut-sets with respect c,b and d,b in G, then $C_{ab} = C_{cb} \cup C_{db}$ is a cut-set with respect to a,b in G, provided that a proper sub-set of this union of cut-sets is not a cut-set C_{ab}. Inspection of G_1 and G_2 gives $\{C_{cb}\} = \{c, b, ef, fi, gi, ijh, ijk\}$ and $\{C_{db}\} = \{d, b, fg, gi, fhj, fjk, ijh, ijk\}$. Thus, $\{C_{ab}\} = \{a, b, cd, gi, cfg, def, dfi, efg, hij, ijk, cfhj, cfjk, efhj, efjk\}$. By the min-cut max-flow theorem, a flow

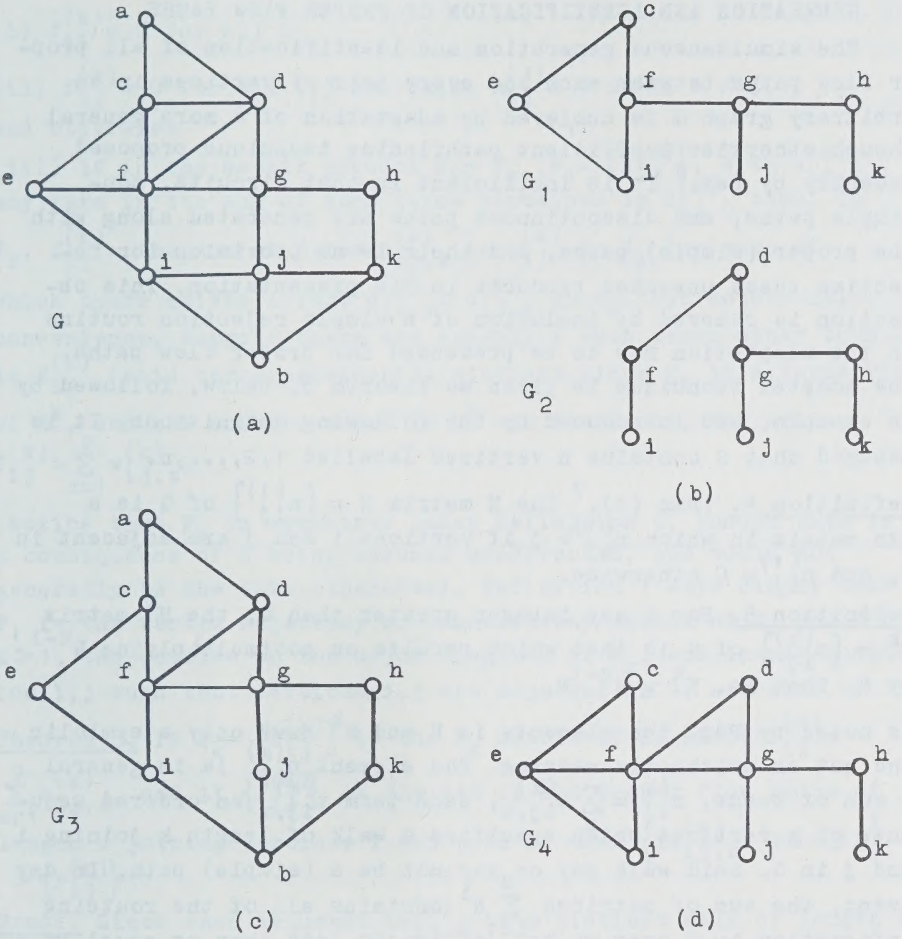

a
c
d
e
f
g
h
i
j
k
G
b
(a)
c
e
f
g
h
G1
i
j
k
d
f
g
h
G2
i
j
k
(b)
a
c
d
e
f
g
h
i
j
k
G3
b
(c)
c
d
e
f
g
h
G4
i
j
k
(d)

Figure 1.

f_{ab} is attainable between vertices a and b in G if and only if $f_{ab} \leq \min|C_{ab}|$ over all $C_{ab} \in \{C_{ab}\}$.

In the next section, a pathfinding technique is given by which the set of all proper flow paths between each and every pair of vertices in G is generated simultaneously, rather than between one pair at a time as in the above example. It is of practical importance to be able to do this in multi-commodity simultaneous-flow applications. The only knowledge required of G is its incidence relationships.

GENERATION AND IDENTIFICATION OF PROPER FLOW PATHS

The simultaneous generation and identification of all proper flow paths between each and every pair of vertices in an arbitrary graph G is achieved by adaptation of a more general though otherwise inefficient pathfinding technique proposed recently by Paz.[7] It is inefficient in that circuits, non-simple paths, and discontinuous paths are generated along with the proper (simple) paths, and their is no provision for rejecting these unwanted products in his presentation. This objection is removed by inclusion of a simple rejection routine in the adaptation now to be presented for proper flow paths. The adapted technique is given as Theorem 5, below, followed by an example, and introduced by the following definitions. It is assumed that G contains n vertices labelled 1,2,...,n.

Definition 4. (Paz (1).[7] The N matrix $N = \left[n_{ij}^{(1)}\right]$ of G is a nxn matrix in which $n_{ij}^{(1)} = j$ if vertices i and j are adjacent in G, and $n_{ij}^{(1)} = 0$ otherwise.

Definition 5. For k any integer greater than 0, the N^k matrix $N^k = \left[n_{ij}^{(k)}\right]$ of G is that which results on postmultiplying N^{k-1} by N. That is, $N^k = N^{k-1}N$.

As noted by Paz, the elements in N and N^k have only a symbolic and not an arithmetic meaning. The element $n_{ij}^{(k)}$ is in general a sum of terms, $n_{ij}^{(k)} = \sum_x t_{ij,x}^{(k)}$, each term $t_{ij,x}^{(k)}$ an ordered sequence of k vertices which specifies a walk of length k joining i and j in G. Said walk may or may not be a (simple) path. In any event, the sum of matrices $\sum_{i=1}^{m} N^i$ contains all of the routeing information in G over walks of length less than or equal to m.

Definition 6. If, in N^k, there exists a term $t_{ij,x}^{(k)}$ in the sum of terms $n_{ij}^{(k)}$ and a term $t_{ji,y}^{(k)}$ in $n_{ji}^{(k)}$ such that $jt_{ji,y}^{(k)} = it_{ij,x}^{(k)}$, then $t_{ij,x}^{(k)}$ is said to be equal to $t_{ji,y}^{(k)}$. That is, $t_{ij,x}^{(k)} = t_{ji,y}^{(k)}$.

In other words, Definition 6 says that any pair of terms which are symmetric with respect to the major diagonal in N^k and which specify the same walk, are said to be equal.

Definition 7. The F_k matrix $F_k = \left[f_{ij}^{(k)}\right]$ of G, k any nonzero positive integer, is a nxn matrix such that (a) $F_1 = N$, the N matrix of G and (b), for $k > 1$, F_k has the following properties:

(i) $f_{ii}^{(k)} = 0$ for all i

(ii) $f_{ij}^{(k)} = 0$ for all i, j for which $n_{ij}^{(1)} \neq 0$ in N, $i \neq j$,
and otherwise
(iii) if in the matrix product $H_k = F_{k-1}F_1 = \left[h_{ij}^{(k)}\right]$, with $u_{ij,x}^{(k)}$ any term in the sum of such terms contained in $h_{ij}^{(k)}$, then, in F_k, $f_{ij}^{(k)}$ = the sum of terms $u_{ij,x}^{(k)}$ in $h_{ij}^{(k)}$ in H_k for each of which there exists a term $u_{ji,y}^{(k)} = u_{ij,x}^{(k)}$ in H_k. For notational convenience, suppose there are exactly p such terms other than 0 in $f_{ij}^{(k)}$ (said terms necessarily distinct since H_k is a submatrix of N^k, and the nonzero terms in N^k are distinct), and write $f_{ij}^{(k)} = \sum_{z=1}^{p} w_{ij,z}^{(k)}$.

(Notice that F_k is symmetric under Definition 6, though this is a consequence of G being assumed nondirected, and would not generally be the case otherwise). Definition 7 says simply that F_1 is the vertex adjacency or connectivity matrix of G. And for $k > 1$, the entries on the major diagonal of F_k, and in any position i, j such that vertices i, j are adjacent in G, are zero.

Theorem 5. If $F_k = \left[f_{ij}^{(k)}\right]$ is the F_k matrix of G, where $f_{ij}^{(k)} = \sum_{z=1}^{p} w_{ij,z}^{(k)}$, and if $\left\{P_{ij}^{(k)}\right\}$ is the set of all proper flow paths of length k joining vertices i and j in G, then $\left\{iw_{ij,z}^{(k)}:\ 1 \leq z \leq p\right\} = \left\{P_{ij}^{(k)}\right\}$.

Proof. Since each sequence $iw_{ij,z}^{(k)}$ is a distinct walk of length k joining the vertices i, j in G, it suffices to prove that each such walk is a proper flow path $P_{ij}^{(k)}$ in G and, conversely, that corresponding to each such $P_{ij}^{(k)}$ in G there exists a term $w_{ij,z}^{(k)}$ in $f_{ij}^{(k)}$. Since $F_1 = N$, the theorem is certainly true for k = 1. For $k > 1$, $f_{ij}^{(k)}$ is by definition zero whenever i=j and whenever i and j are adjacent in G. In particular, for k = 2 each walk $iw_{ij,z}^{(2)}$ thus contains three distinct vertices, its terminal vertices are not adjacent in G, and it is therefore a proper flow path $P_{ij}^{(2)}$ of length 2 in G. Conversely, each proper flow path $P_{ij}^{(2)}$ of length 2 in G appears exactly once in $N^2 = F_1^2$, and is admissible in F_2. Thus, the theorem is true for k = 2.
The proof is completed by induction on k, as follows. Assume it is true for k = m-1, so that each walk $iw_{ij,z}^{(m-1)}$, $w_{ij,z}^{(m-1)}$ in $f_{ij}^{(m-1)}$,

is a distinct proper flow path $P_{ij}^{(m-1)}$ of length m-1 in G and, conversely, corresponding to each $P_{ij}^{(m-1)}$ of length m-1 in G there exists a term $w_{ij,z}^{(m-1)}$ in $f_{ij}^{(m-1)}$. By the definition of F_m, each term $w_{ik,z}^{(m)}$ in $f_{ik}^{(m)}$ is the product of some term $w_{ij,z}^{(m-1)}$ in $f_{ij}^{(m-1)}$ and a vertex k which is adjacent to the vertex j in G. That is, $w_{ik,z}^{(m)}$ in F_m is equal to $w_{ij,z}^{(m-1)}k$. By the properties of $f_{ij}^{(k)}$, $k>1$, given earlier, the specific vertices i, j, k which appear as subscripts in $w_{ij,z}^{(m-1)}$ and $w_{ik,z}^{(m)}$ are distinct in G, and neither the pair i, j nor the pair i, k are adjacent in G. Since $iw_{ij,z}^{(m-1)}$ is a proper flow path, then $iw_{ik,z}^{(m)}$ is also a proper flow path if and only if k is not adjacent in G to any internal vertex in $w_{ij,z}^{(m-1)}$. Suppose, however, that k and the internal vertex v in $w_{ij,z}^{(m-1)}$ are adjacent in G. Then the sequence $kw_{vk,z}^{(x)}$, $x<m$, is not a proper flow path. Which, since $x<m$, contradicts the induction hypothesis. Hence, each walk $iw_{ij,z}^{(m)}$ of length m is a distinct proper flow path $P_{ij}^{(m)}$ in G. Conversely, since each proper flow path $P_{ij}^{(m)}$ of length m in G is admissible in F_m, then there exists a unique term $w_{ij,z}^{(m)}$ in $f_{ij}^{(m)}$ corresponding to $P_{ij}^{(m)}$. Which completes the proof.

The mechanics involved in generating the set of all proper flow paths is illustrated by applying Theorem 5 to the graph G_4 of Figure 1(d), which is a subgraph of G in Figure 1(a). The F_k matrices of G_4 are given below. For illustration in forming F_k from $H_k = F_{k-1}F_1$, the stroked entries shown in F_3 are all of those which belong to H_3 but not to F_3. Note that $F_k \equiv 0$ for all $k>4$. Thus, the longest proper flow path in G_4 is of length 4.

The F_k Matrices of the Graph G_4 of Figure 1(d).

k=1	c	d	e	f	g	h	i	j	k
c	-	-	e	f	-	-	-	-	-
d	-	-	-	f	g	-	-	-	-
e	c	-	-	f	-	-	i	-	-
f	c	d	e	-	g	-	i	-	-
g	-	d	-	f	-	h	-	j	-
h	-	-	-	-	g	-	-	-	k
i	-	-	e	f	-	-	-	-	-
j	-	-	-	-	g	-	-	-	-
k	-	-	-	-	-	h	-	-	-

k=2	c	d	e	f	g	h	i	j	k
c	-	fd	-	-	fg	-	ei+fi	-	-
d	fc	-	fe	-	-	gh	fi	gj	-
e	-	fd	-	-	fg	-	-	-	-
f	-	-	-	-	-	gh	-	gj	-
g	fc	-	fe	-	-	-	fi	-	hk
h	-	gd	-	gf	-	-	-	gj	-
i	ec+fc	fd	-	-	fg	-	-	-	-
j	-	gd	-	gf	-	gh	-	-	-
k	-	-	-	-	hg	-	-	-	-

k=3	c	d	e	f	g	h	i	j	k
c	-	~~fgd~~	-	-	~~fdg~~	fgh	-	fgj	-
d	~~fec~~	-	~~fce+fie~~	-	-	-	~~fei~~	-	ghk
e	-	~~fgd~~	-	-	~~fdg~~	fgh	-	fgj	-
f	-	-	-	-	-	-	-	-	ghk
g	~~fec~~	-	~~fce+fie~~	-	-	-	~~fei~~	-	-
h	gfc	~~gfd~~	gfe	~~gdf~~	-	-	gfi	-	-
i	-	~~fgd~~	-	-	~~fdg~~	fgh	-	fgj	-
j	gfc	~~gfd~~	gfe	~~gdf~~	-	-	gfi	-	ghk
k	-	hgd	-	hgf	-	-	-	hgj	-

In F_4 the only nonzero entries are $f_{ck}^{(4)} = f_{ek}^{(4)} = f_{ik}^{(4)} = fghk$, $f_{kc}^{(4)} = hgfc$, $f_{ke}^{(4)} = hgfe$, and $f_{ki}^{(4)} = hgfi$. And $F_k \equiv 0$ for all $k > 4$.

Per Theorem 5, each and every proper flow path of length k in G_4 is uniquely present in $f_{ij}^{(k)}$ and in $f_{ji}^{(k)}$ in F_k. For example, the set union $\bigcup_{\substack{r=c,d\\ s=i,j,k}} \{P_{rs}\}$ of all proper flow paths between $r = c,d$ and $s = i,j,k$ in G_4 is seen from F_1, F_2, F_3, F_4 to be $\bigcup_{\substack{r=c,d\\ s=i,j,k}} \{P_{rs}\} = \{cei, cfi, dfi, dgj, cfgj, dghk, cfghk\}$. Since vertices e,f do not appear together in any of these paths, then edge (e,f) in G_4 is redundant with respect to this set of paths. Which confirms the same finding made earlier by inspection, this set of paths being those which remain on erasing vertices a,b, with their incident edges, in the graph G_3 of Figure 1(c).

REFERENCES

1. B.R. Myers and M.A. Tapia, Analysis and synthesis of node-weighted networks, Proc. Conference on Electrical Networks, University of Newcastle upon Tyne, England, 1966.
2. M.A. Tapia, Analysis and synthesis of weighted communication networks, Ph.D. thesis, University of Notre Dame, 1966.
3. B.R. Myers, Analysis and synthesis of weighted communication networks, Proc. NATO Advanced Study Institute on Network and Switching Theory, Trieste, Italy, 1966; Academic Press, Inc.
4. S.S. Yau, Synthesis of radio-communication nets, IRE Trans., CT-9, 62-68, 1962.
5. S.L. Hakimi, Recent progress and new problems in applied graph theory, Proc. IEEE Region VI Annual Conference, Tucson, Arizona, 1966.
6. E.A. Davila, Simultaneous flows in node-weighted networks, Ph.D. thesis, University of Notre Dame, 1967.
7. I.M. Paz, Pathfinding through a communication network, Proc. IEE, 1, 76-78, 1967.

ON THE CHARACTERIZATION OF NONUNIFORM LOSSLESS TRANSMISSION LINES OF FINITE LENGTH

M. WOHLERS
Grumman Aircraft Engineering Corporation
Bethpage, New York 11714

ABSTRACT

This paper concerns the terminal behavior of nonuniform lossless transmission lines of finite length, specifically the properties of their scattering parameters. It is shown that the input reflection factor (S_{11} with the normalization used) of a smooth line of known electrical length, terminated in a known passive load, uniquely determines the local characteristic impedance and the other scattering parameters of the line. A formal realizability theory is obtained in terms of the input reflection factor with resistive termination. The technique underlying this theory involves the reduction of the problem to the inversion of a Sturm-Liouville equation; a classic problem for which a well developed theory is available. This latter theory also suggests a possible synthesis procedure.

INTRODUCTION

A complete theory is available for networks composed of lossless uniform transmission lines all with the same electrical length. Except for specific examples, however, no comparable theory exists for the so-called nonuniform lines whose local characteristic impedance varies with position along the line. A basic difficulty is that the differential equations describing nonuniform lines have variable coefficients and their solutions cannot be obtained in closed form. Various techniques have been described for the analysis of such lines, some of which utilize a Volterra integral equation formulation. The advantage of this formulation is that it allows solutions to be obtained in a systematic manner by successive approximations. To date, however, no formal realizability theory for nonuniform lines had been developed. The purpose of this paper is to present a theory applicable to "smooth" lines, i.e., lines whose local characteristic impedance is at least twice continuously differentiable.

A REALIZABILITY THEORY

The problem may be stated as follows: Given the input reflection factor of a smooth nonuniform lossless transmission line of electrical length Δ, terminated in some given passive impedance Z_Δ, can one find the internal description of the line $\left(\text{i.e., its local characteristic impedance } Z_o(y) = \sqrt{L(y)/C(y)}\right)$ as a function of position? The answer is yes, and the first step in demonstrating it is to show that the following theorem is true.

Theorem 1 [Ref. 1]. If $S(p)$ is the input reflection factor of a "smooth" lossless line of electrical length Δ terminated in a "regular" passive impedance $Z_\Delta(p)$, then $S(p)$ uniquely determines that scattering matrix of the line which is normalized to 1Ω at port one and $Z_\Delta(p)$ at port two.

By a "regular" passive impedance $Z_\Delta(p)$ we shall mean those meromorphic positive-real functions such that the factorization $Z_\Delta(p) + Z_\Delta(-p) = 2h_\Delta(p)h_\Delta(-p)$ may be performed so that $h_\Delta(p)$ is meromorphic, $h_\Delta(p)$ and $h_\Delta(p)/h_\Delta(-p)$ are bounded and analytic in $\text{Re } p \geq 0$, $h_\Delta(p)$ is nonzero in $\text{Re } p < 0$, and

$$\lim_{|p| \to \infty} \frac{\log h_\Delta(p)}{p} = 0$$

in $\text{Re } p > 0$. Moreover, one may find this scattering matrix explicitly, thus

Corollary 1 [Ref. 1]. Assume that the conditions of Theorem 1 apply and define $A(p)$ to be that unique factorization of

$$1 - S(p)S(-p) = A(p)A(-p)$$

which is analytic and nonzero in $\mathrm{Re}\, p > 0$, satisfying

$$\lim_{|p| \to \infty} \frac{\log A(p)}{p} = 0 \quad \text{in} \quad \mathrm{Re}\, p > 0 .$$

Then the scattering parameters of the line (normalized to Z_Δ at $y = \Delta$ and 1Ω at $y = 0$) are given by $S_{11}(p) = S(p)$, $S_{12}(p) = S_{21}(p) = A(p)b(p)\, e^{-\Delta p}$ and $S_{22}(p) = -\dfrac{S_{11}(-p)S_{12}(p)}{S_{12}(-p)}$, where $b(p)$ is the regular Blaschke product or all-pass function defined by the zeros of $h_\Delta(p)$ in $\mathrm{Re}\, p > 0$.

On the basis of this theorem one may conclude that if $S(p)$, Δ, and $Z_\Delta(p)$ are given, we can find the input impedance of the line with an arbitrary terminating impedance. In particular, by letting $Z_\Delta(p) = pK$ (an inductor), we may relate the problem to a consideration of a Sturm-Liouville system. Specifically one has

Theorem 2 [Ref. 1]. Every p for which a nonzero solution exists to

$$\frac{d^2u}{dy^2} - Qu - p^2u = 0$$

that satisfies the boundary conditions,

$$\frac{du}{dy}(0) + u(0)\left[\frac{\frac{1}{2}\frac{dZ_o}{dy}(0)}{Z_o(0)}\right] = 0$$

$$\frac{du}{dy}(\Delta) + u(\Delta)\left[\frac{\frac{1}{2}\frac{dZ_o}{dy}(\Delta)}{Z_o(\Delta)} + \frac{Z_o(\Delta)}{K}\right] = 0$$

where

$$Q = \sqrt{Z_o(y)}\,\frac{d^2\left(1/\sqrt{Z_o(y)}\right)}{dy^2} \quad \text{and} \quad Z_o(y) > 0 ,$$

is a pole of the input impedance, $Z_{in}(p)$, of a "smooth" lossless line of length Δ with local characteristic impedance $Z_o(y)$, terminated in $Z_\Delta = pK$, and conversely.

Thus, the set of poles of $Z_{in}(p)$, corresponding to an inductive terminating impedance, constitutes a set of eigenvalues for the Sturm-Liouville system described by the differential equation $d^2u/dy^2 - Qu - p^2u = 0$ subject to the boundary conditions $\frac{du}{dy}(0) + hu(0) = 0$ and $\frac{du}{dy}(\Delta) + Hu(\Delta) = 0$. Such systems have been studied extensively by mathematicians. By using the work of V. A. Marčenko [Ref. 2] and of I. M. Gelfand and B. M. Levitan [Ref. 3] on the inversion of a Sturm-Liouville system, it is possible to obtain the following realizability theorem when we assume the lines are "infinitely smooth":

Theorem 3. If $S(p)$ is to be the input reflection factor (normalized to 1) of a nonuniform lossless transmission line of length Δ, terminated in 1Ω, then the following conditions are both necessary and sufficient if the local characteristic impedance $Z_o(y)$ is positive and infinitely differentiable:

1) $S(p)$ is a meromorphic function, analytic and bounded by one in $\text{Re } p \geq 0$, satisfying $S(\bar{p}) = \overline{S(p)}$;

2) $S(0) = 0$;

3) $1 - S(p)S(-p)$ is nonzero in the finite p-plane;

4) the poles of $1 - S(p)S(-p)$ are identical (including order) to the poles of $S(p)$ in $\text{Re } p < 0$; and

5) if $A(p)$ is that factorization of $1 - S(p)S(-p) = A(p)A(-p)$ such that $A(p)$ is analytic and nonzero in

$\operatorname{Re} p \geq 0$, and $\lim\limits_{|p| \to \infty} \frac{\log A}{p} = 0$ in $\operatorname{Re} p > 0$, then

a) for $|p| \to \infty$

$$\frac{1 \pm S(p)}{A(p)\, e^{-\Delta p}} = e^{\Delta p}\left[c_{\pm} + 0\left(\frac{1}{p}\right)\right] + e^{-\Delta p}\left[d_{\pm} + 0\left(\frac{1}{p}\right)\right]$$

where $c_{\pm} > 0$, and $\left|\frac{d_-}{c_-}\right| < 1$, and

b) for $|\omega| \to \infty$

$$\frac{1 - S(j\omega)}{A(j\omega)\, e^{-j\omega\Delta}} = e^{j\omega\Delta}\left[c_- + \frac{c_1}{\omega} + \ldots + \frac{c_n}{\omega^n} + \ldots\right]$$

$$+ e^{-j\omega\Delta}\left[d_- + \frac{d_1}{\omega} + \ldots + \frac{d_n}{\omega^n} + \ldots\right],$$

where for any n, c_n and d_n are constants.

Moreover, the proof of the theorem also allows one to conclude

Theorem 4 [Ref. 1]. The input reflection factor of a "smooth" lossless line of a given electrical length terminated in a given "regular" passive impedance uniquely determines the local characteristic impedance of the line as a function of position.

Finally, one may develop an algorithm for determining the local characteristic impedance $Z_o(y)$ by using a constructive procedure due to Marčenko for inverting a Sturm-Liouville system, i.e., finding $Q(y)$, given two sets of eigenvalues corresponding to two sets of boundary conditions (h, H_1) and (h, H_2) with $H_1 \neq H_2$ [Refs. 1 and 2]. Thus on the basis of Theorem 2, we may find two such sets as the poles of the input impedance of the line with two different inductive terminations. Specifically, from Corollary 1 we see that these poles can be determined from $S(p)$, $Z_\Delta(p)$ and Δ, i.e., from the input reflection factor with an arbitrary "regular" passive termination. Unfortunately, the inversion procedure due to Marčenko cannot be carried out in closed form and one must resort to infinite series expansions. Thus from a practical standpoint the synthesis problem remains essentially unsolved.

REFERENCES

1. M. R. Wohlers, "A Realizability Theory for Smooth Lossless Transmission Lines," IEEE Trans. on Circuit Theory, Vol. CT-13, December 1966.
2. V. A. Marčenko, "Some Questions of the Theory of One Dimensional Linear Differential Operators of the Second Order," I, II, Trudy Moskov. Mat. Obšč., 1 (1952), 327-420; 2 (1953), 3-83 (in Russian).
3. I. M. Gelfand and B. M. Levitan, "On the Determination of a Differential Equation from its Spectral Function," Izv. Akad. Nauk SSSR Ser. Mat., 15 (1951), 309-350 (in Russian); English transl., Amer. Math. Soc. Transl. (2) 1 (1955), 253-304.

FEEDBACK CONTROL WITH MAGNITUDE CONSTRAINTS FOR SYSTEMS WITH UNKNOWN PARAMETERS

R. A. WERNER
TRW Systems
Redondo Beach, California

ABSTRACT

In this paper a feedback control law satisfying magnitude constraints is described that is near optimum for controlling a system that contains parameters that are unknown but constant. The system considered is linear in the state variables, but can be nonlinear in the parameters. The feedback control law is based on a linear perturbation from the nominal optimal trajectory.

I. INTRODUCTION

A method for solving optimal control problems with magnitude constraints on the control is given in [1]. Linear perturbation techniques have been successfully used for solving optimal control problems with no constraints on the control function [2], [3]. The purpose of this paper is to derive a near optimal control law satisfying magnitude constraints using linear perturbation techniques.

In Section 2, the complete formulation of the problem is presented. In Section 3, the proposed solution for determining a near optimal control law for systems with unknown parameters is given. The proposed solution is identical to the form of the control law as given in [1]. Finally, in Section 4, a numerical solution is presented indicating the effectiveness of the proposed solution.

II. STATEMENT OF THE PROBLEM

This paper considers a dynamical system described by

$$\dot{\underline{x}} = A(\underline{v})\,\underline{x} + B(\underline{v})\,\underline{u} + E(\underline{v}) \tag{1}$$

the output of which is given by

$$\underline{y} = C(\underline{v})\,\underline{x} + F(\underline{v}) \tag{2}$$

with the initial state of the system at t = 0

$$\underline{x}(0) = \underline{x}_o\,(\underline{v}) \tag{3}$$

and where $\underline{x}$ is an n_x-rowed state vector, $\underline{y}$ is an n_y output vector, $\underline{u}$ is an n_u control vector, and $\underline{v}$ is an n_v-vector of unknown parameters, assumed constant. The matrices A, B, C, E and F are of the appropriate dimensions and are assumed to be known rational functions of the parameter vector $\underline{v}$.

For any time $t \geq 0$, the control $\underline{u}$ is constrained to lie in the closed set U in the n_u dimensional Euclidean space and to be piecewise continuous and bounded in the interval $[0, t_f]$. Any control satisfying these constraints is called an admissible control.

Consider next a cost function which is to be minimized with respect to $\underline{u}(t)$

$$J = \int_{o}^{t_f} f_o(\underline{x}, \underline{u}, \underline{v})\, dt \tag{4}$$

where f_o is a rational function of $\underline{v}$ and a polynomial function of $\underline{x}$ and $\underline{u}$.

To simplify notation, let us write (1) as

$$\dot{\underline{x}} = \underline{f}(\underline{x}, \underline{u}, \underline{v}) \tag{5}$$

or in component form

$$\dot{x}_i = f_i(\underline{x}, \underline{u}, \underline{v}), \ i = 1, 2, \ldots, n_x \tag{6}$$

The problem solved by Pontryagin in our context is the following: Let $\underline{v}$ be known. Determine an admissible control function $\underline{u}^o(t)$ such that J as specified by (4) is minimized subject to (5) and (3).

In Pontryagin's method, the Hamiltonian H is defined by

$$H(\underline{\lambda}, \underline{x}, \underline{u}, \underline{v}) = -f_o + \underline{\lambda}^T \underline{f} \tag{7}$$

where $\underline{\lambda}(t)$ is the n_x-vector of adjoint (auxiliary) variables that must satisfy the linear differential equation

$$\dot{\underline{\lambda}} = -\mathrm{grad}_{\underline{x}} H \tag{8}$$

for $0 \leq t \leq t_f$ for any admissible control function $\underline{u}(t)$.

If $\underline{u}^o(t)$ is the solution to this problem, and if $\underline{x}^o(t)$ and $\underline{\lambda}^o(t)$ are the corresponding state and adjoint variable solutions, respectively, then Pontryagin's maximum principle states that a necessary condition that $\underline{u}^o(t)$ must satisfy is

$$H(\underline{\lambda}^o, \underline{x}^o, \underline{u}^o, \underline{v}) = \max_{u \epsilon U} H(\underline{\lambda}^o, \underline{x}^o, \underline{u}, \underline{v}) \tag{9}$$

for $0 \leq t \leq t_f$.

The problem solved in this paper can now be stated. Assume that an optimal solution has been obtained for the system governed by (3), (5) and cost function (4) with $\underline{v} = \underline{0}$. This is called the nominal solution. (We can assume without loss of generality that the nominal value of $\underline{v}$ is $\underline{0}$). The problem is to determine a perturbed control which is near optimal for small parameter changes, $\underline{v} \neq \underline{0}$, and which is a function of the output as specified by (2).

The solution to the above problem will be presented in the following section. Before proceeding further, however, let us define the set $T(\underline{v})$. As stated above, the control was constrained to be at least piecewise continuous. Let $t_1, t_2, \ldots, t_{m-1}$ be switching times of the control, where $0 < t_1, < t_2, < \ldots, < t_{m-1} < t_f$, i. e., $\underline{u}^o(t)$ is not continuously differentiable at $t = t_i$ for $i = 1, \ldots, m-1$. $T(\underline{v})$ is the set of all switching times for any given $\underline{v}$; i. e.,

$$T(\underline{v}) = \left\{ t_j(\underline{v}) \;\middle|\; j = 1, 2, \ldots, m-1 \right\} \tag{10}$$

(We also define $t_o = 0$ and $t_m = t_f$).

We will assume that in some neighborhood of $\underline{v} = \underline{o}$

$$\frac{dt_j(\underline{v})}{d\, v_i}, \quad \underline{u}_a(t_j^-(\underline{v}), \underline{v}), \quad \text{and} \quad \underline{u}_a(t_j^+(\underline{v}), \underline{v})$$

exist and are continuous, and

$$\frac{\partial}{\partial v_i} \underline{u}_a(t, \underline{v})$$

exists and is bounded for t in the complement of the set $T(\underline{v})$, where for any $\underline{v}$, $\underline{u}_a(t, \underline{v}) = \underline{u}^o(t)$ for $0 \leq t \leq t_f$.[*] This is the case for a large class of systems.

We also assume that the $n_y \times n_v$ matrix

$$\left[\frac{\partial \underline{y}_a(t, \underline{v})}{\partial \underline{v}} \right]_{\underline{v}=\underline{o}}$$

exists and has rank n_v almost everywhere on $[0, t_f]$, where $\underline{y}_a(t, \underline{v})$ is the output of the system corresponding to $\underline{u} = \underline{u}_a(t, \underline{v})$[**]. This requires that the system be parameter observable when $\underline{u} = \underline{u}_a$. This is also the case for a large class of systems.

III. SOLUTION

The solution to the above stated problem consists of essentially two important steps using prestored optimal solution data for $\underline{v}=\underline{o}$. 1) Determine the parameter $\underline{v}$ from the observation $\underline{y}$. 2) Determine the admissible control $\underline{u}_a$ as a function of the parameter $\underline{v}$.

In step 1), called the identification problem, $\underline{v}$ is approximated by a function $\underline{v}_c(t, \underline{y})$ chosen so that when $\underline{y} = \underline{y}_a(t, \underline{v})$ (corresponding to $\underline{u} = \underline{u}_a(t, \underline{v})$) $\underline{v}_c(t, \underline{y}_a(t, \underline{v}))$ is a first order approximation of $\underline{v}$ for $0 < t \leq t_f$. This function $\underline{v}_c$ turns out to be a linear function of $\underline{y}$.

---

[*] We define t_j^- and t_j^+ as follows: for any arbitrarily small $\epsilon > 0$

$$t_j > t_j^- > t_j - \epsilon, \text{ and } t_j < t_j^+ < t_j + \epsilon$$

[**] This requires that $n_y \geq n_v$. However, it can be shown [2] that n_y can sometimes be increased by forming new outputs that are functionals (time integrals or time derivations) of the original output defined by (2).

In Step 2), called the control problem, $\underline{u}$ is expressed as a nonlinear, time-varying function of $\underline{v}$ chosen so that it is a first order approximation of $\underline{u}_a(t, \underline{v})$ in terms of $\underline{v}$. It is chosen so that when $\underline{v}_c(t, \underline{y})$ is substituted for $\underline{v}$ in this control function, the resulting composite function is $\underline{u}_c(t, \underline{y})$, the desired feedback control function.

First, the system identification problem will be solved. In order to determine $\underline{v}_c(t, \underline{y})$, it is first necessary to find an $n_v \times n_y$ matrix $D(t)$, $0 \le t \le t_f$, such that if

$$\underline{w}_a(t, \underline{v}) \triangleq D(t)\, \underline{y}_a(t, \underline{v}) \tag{11}$$

then the $n_v \times n_v$ matrix

$$\left(\frac{\partial \underline{w}_a}{\partial \underline{v}}\right)_{\underline{v}=\underline{o}} \tag{12}$$

is nonsingular almost everywhere on $[0, t_f]$. The fact that this can be done is implied by the assumption that the system is parameter observable.

The function $\underline{v}_c(t, \underline{y})$ can be defined by

$$\underline{v}_c(t, \underline{y}) = S^j(t)\,[\underline{y} - \underline{y}^j_a(t, \underline{0})] \tag{13}$$

where

$$S^j(t) = \left[\frac{\partial}{\partial \underline{v}} \underline{w}_a(t, \underline{v})\right]^{\substack{-1 \\ j}}_{\underline{v}=\underline{o}} D(t) \tag{14}$$

wherever the inverse exists for $t_{j-1} < t \le t_j$, $j = 1, \ldots, m \cdot$ ($\underline{v}_c = \underline{v}_c|_{t=t^-}$ wherever the inverse does not exist). With $\underline{v}_c$ defined this way, if $\underline{u} = \underline{u}_a$ so that $\underline{y} = \underline{y}_a$, it is easy to show that $\underline{v}_c(t, \underline{y})$ is the desired first order approximation of $\underline{v}$ on $(0, t_f]$.

Now that the solution of the identification problem has been described, the control problem will be solved. The control is expressed as

$$\underline{u} = \underline{p}(\underline{\lambda}(t, \underline{v}),\ \underline{x}(t, \underline{v}),\ \underline{v}) \tag{15}$$

where

$$\underline{\lambda}(t, \underline{v}) = \underline{\lambda}^j_a(t, \underline{0}) + \left[\frac{\partial}{\partial \underline{v}}\ \underline{\lambda}_a(t, \underline{v})\right]^j_{\underline{v}=\underline{o}} \underline{v} \tag{16}$$

and

$$\underline{x}(t, \underline{v}) = \underline{x}^j_a(t, \underline{0}) + \left[\frac{\partial}{\partial \underline{v}}\ x_a(t, \underline{v})\right]^j_{\underline{v}=\underline{o}} \underline{v} \tag{17}$$

for $t_{j-1} < t \le t_j$, $j = 1, \ldots, m$. Observe that $\underline{\lambda}(t, \underline{v})$ and $\underline{x}(t, \underline{v})$ are first order approximations of $\underline{\lambda}_a(t, \underline{v})$ and $\underline{x}_a(t\ \underline{v})$* in terms of $\underline{v}$. The function $\underline{p}$ is chosen so that

$$\underline{u}_a(t, \underline{v}) = \underline{p}(\underline{\lambda}_a(t, \underline{v}), \underline{x}_a(t, \underline{v}), \underline{v}) \tag{18}$$

for every $\underline{v} \in V$ and for $0 \le t \le t_f$. Since $\underline{\lambda}(t, \underline{v})$ and $\underline{x}(t, \underline{v})$ in Equations (16) and (17) approximate $\underline{\lambda}_a(t, \underline{v})$ and $\underline{x}_a(t, \underline{v})$, respectively, $\underline{p}(\underline{\lambda}, \underline{x}, \underline{v})$ approximates $\underline{u}_a(t, \underline{v})$ for $\underline{v}$ sufficiently small.

The system identification and control solutions can be combined to give a nearly optimal feedback control of the form

$$\underline{u}_c(t, \underline{y}) = \underline{p}(\underline{\lambda}(t, \underline{v}_c), \underline{x}(t, \underline{v}_c), \underline{v}_c) \tag{19}$$

where $\underline{\lambda}(t, \underline{v})$ and $\underline{x}(t, \underline{v})$ are given in Equations (16) and (17), and $\underline{v}_c = \underline{v}_c(t, \underline{y})$ is given in Equation (13). Combining these equations, one gets

$$\underline{\lambda}(t, \underline{v}_c) = \underline{\lambda}_a^j(t, \underline{0}) + G^j(t)\left[\underline{y} - \underline{y}_a^j(t, \underline{0})\right] \tag{20}$$

and

$$\underline{x}(t, \underline{v}_c) = \underline{x}_a^j(t, \underline{0}) + K^j(t)\left[\underline{y} - \underline{y}_a^j(t, \underline{0})\right] \tag{21}$$

where

$$G^j(t) = \left[\frac{\partial}{\partial \underline{v}}\underline{\lambda}_a(t, \underline{v})\right]^j_{\underline{v}=\underline{o}} \left[\frac{\partial}{\partial \underline{v}}\underline{w}_a(t, \underline{v})\right]^{j\ -1}_{\underline{v}=\underline{o}} D(t) \tag{22}$$

and

$$K^j(t) = \left[\frac{\partial}{\partial \underline{v}}\underline{x}_a(t, \underline{v})\right]^j_{\underline{v}=\underline{o}} \left[\frac{\partial}{\partial \underline{v}}\underline{w}_a(t, \underline{v})\right]^{j\ -1}_{\underline{v}=\underline{o}} D(t) \tag{23}$$

for $t_{j-1} < t \le t_j$, $j = 1, 2, \ldots, m$.

$$\underline{\lambda} = \underline{\lambda}\Big|_{t=t^-} \text{ and } \underline{x} = \underline{x}\Big|_{t=t^-}$$

wherever

$$\left[\frac{\partial \underline{w}_a}{\partial \underline{v}}\right]^{j\ -1}_{\underline{v}=\underline{o}}$$

does not exist.

---

*The functions $\underline{x}_a(t, \underline{v})$ and $\underline{\lambda}_a(t, \underline{v})$ are the state and auxiliary variable solutions, respectively, when $\underline{u} = \underline{u}_a(t, \underline{v})$. Observe that $\underline{x}_a^j(t, \underline{0})$, $\underline{\lambda}_a^j(t, \underline{0})$, and $\underline{y}_a^j(t, \underline{0})$, $j = 1, \ldots, m$ are the optimal nominal solutions which are assumed known.

It can be shown [2] that if $\underline{u}_c$ is a first order approximation of $\underline{u}_a(t, \underline{v})$ in terms of $\underline{v}$ on $(0, t_f)$, then $\underline{y}$ is a first order approximation of $\underline{y}_a$ in terms of $\underline{v}$, and, if $\underline{y}$ is a first order approximation of $\underline{y}_a(t, \underline{v})$ in terms of $\underline{v}$, then $\underline{u}_c(t, \underline{y})$ is a first order approximation of $\underline{u}_a(t, \underline{v})$ in terms of $\underline{v}$ on $(0, t_f)$. Therefore, if $\underline{u} = \underline{u}_c(t, \underline{y})$ is the control for the system, then $\underline{u}$, $\underline{x}$ and $\underline{y}$ are first order approximations of $\underline{u}_a$, $\underline{x}_a$ and $\underline{y}_a$, respectively in terms of v.

Now that a general description has been given of the form of the optimal feedback controller, the method of constructing it must be given. It is still necessary to determine the function $\underline{p}$ appearing in Equation (18) and the time functions $S^j(t)$, $G^j(t)$ and $K^j(t)$ defined in Equations (14), (22) and (23) for $j = 1, \ldots, m$.

In order to determine $\underline{p}$, observe that for any $\underline{v}$, Equation (9) generally leads to a solution of the form

$$\underline{u}^o = \underline{p}^o(\underline{\lambda}^o, \underline{x}^o, \underline{v}) \qquad (24)$$

for some function $\underline{p}^o$. Since for any fixed $\underline{v}$, $\underline{u}_a(t, \underline{v}) = \underline{u}^o(t)$ and hence $\underline{x}_a(t, \underline{v}) = \underline{x}^o(t)$ and $\underline{\lambda}_a(t, \underline{v}) = \underline{\lambda}^o(t)$, therefore Equation (18) is satisfied if $\underline{p} = \underline{p}^o$, and hence $\underline{p}^o$ is the desired function $\underline{p}$.

In order to determine $S^j(t)$, $G^j(t)$ and $K^j(t)$, it is first necessary to find the sensitivity coefficients of the optimal solution. The optimal solution $\underline{u}_a(t, \underline{v})$, $\underline{x}_a(t, \underline{v})$, $\underline{\lambda}_a(t, \underline{v})$, $\underline{y}_a(t, \underline{v})$ and $T(\underline{v})$ must satisfy Equations (1), (2), (3), (8) and (9) for every $\underline{v}$. If, in addition, the final state $\underline{x}_f(\underline{v})$ is given

$$\underline{x}_a(t_f, \underline{v}) = \underline{x}_f(\underline{v}) \qquad (25)$$

for every $\underline{v}$. If the final state is free, then

$$\underline{\lambda}_a(t_f, \underline{v}) = \underline{0} \qquad (26)$$

for every $\underline{v}$. If the final time t_f is free, then

$$H(\underline{\lambda}_a, \underline{x}_a, \underline{u}_a, \underline{v})\Big|_{t=t_f} = 0 \qquad (27)$$

for every $\underline{v}$.

Since Equations (1), (2), (3), (8), (9), (25), (26) and (27) must hold for every $\underline{v} \in V$, the total derivative of both sides of each equation with respect to v_i at $\underline{v} = \underline{0}$ must be equal wherever the derivatives exist for $i = 1, \ldots, n_v$.

Taking the total derivative of both sides of Equation (1) with respect to v_i and reversing the order of differentiation with respect to v_i and t leads to

$$\frac{\partial}{\partial t}\left[\frac{\partial \underline{x}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{o}} = \left[\frac{\partial A}{\partial v_i}\right]_{\underline{v}=\underline{o}} \underline{x}_a^{\,j}(t,\underline{0}) + A(\underline{0})\left[\frac{\partial \underline{x}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{o}}$$

$$+ \left[\frac{\partial B}{\partial v_i}\right]_{\underline{v}=\underline{o}} \underline{u}_a^{\,j}(t,\underline{0}) + B(\underline{0})\left[\frac{\partial \underline{u}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{o}} + \left[\frac{\partial E}{\partial v_i}\right]_{\underline{v}=\underline{o}} \tag{28}$$

for $t_{j-1} < t < t_j$, $j = 1, \ldots, m$. Taking the total derivative of Equation (8) with respect to v_i leads to a similar set of n_x first order differential equations. Taking the total derivative of Equation (25) with respect to v_i gives

$$\left[A\,\underline{x}_a^m(t_{f_a}, \underline{0}) + B\,\underline{u}_a^m(t_{f_a}, \underline{0}), + E\right]_{\underline{v}=\underline{o}} \left[\frac{\partial t_{f_a}}{\partial v_i}\right]_{\underline{v}=\underline{o}} +$$

$$+ \left[\frac{\partial \underline{x}_a}{\partial v_i}\right]^m_{\substack{t=t_{f_a}\\ \underline{v}=\underline{o}}} = \left[\frac{\partial \underline{x}_f}{\partial v_i}\right]_{\underline{v}=\underline{o}} \tag{29}$$

Equations (3), (26) and (27) lead to similar sets of boundary conditions. In order to find the boundary conditions of

$$\left[\frac{\partial \underline{x}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{o}}$$

at $t = t_j(\underline{0})$, $j=1, 2, \ldots, m-1$, one must use

$$\underline{x}_a(t_j^-(\underline{v}), \underline{v}) = \underline{x}_a(t_j^+(\underline{v}), \underline{v}) \tag{30}$$

for every $\underline{v}$. Therefore, taking the derivative of Equation (30) with respect to v_i leads to

$$\left[A\,\underline{x}_a + B\,\underline{u}_a + E\right]_{\substack{t=t_j^-\\ \underline{v}=\underline{o}}} \left[\frac{\partial t_j}{\partial v_i}\right]_{\underline{v}=\underline{o}} + \left[\frac{\partial \underline{x}_a}{\partial v_i}\right]^j_{\substack{t=t_j^-\\ \underline{v}=\underline{o}}} = \left[A\,\underline{x}_a + B\,\underline{u}_a - E\right]_{\substack{t=t_j^+\\ \underline{v}=\underline{o}}}$$

$$\left[\frac{\partial t_j}{\partial v_i}\right]_{\underline{v}=\underline{o}} + \left[\frac{\partial \underline{x}_a}{\partial v_i}\right]^{j+1}_{\substack{t=t_j^+\\ \underline{v}=\underline{o}}} \tag{31}$$

for j = 1, 2, ..., m-1. Analogous boundary conditions are found for

$$\left[\frac{\partial \underline{\lambda}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{0}}$$

at $t = t_j(\underline{0})$, j = 1, ..., m. Observe that after the nominal solution is substituted into Equations (28) through (31) and into the equations found by taking the total derivative of Equations (3), (8), (26) and (27) with respect to v_i, one has a set of linear differential equations with linear boundary conditions. These can easily be solved to obtain

$$\left\{ \left[\frac{\underline{x}_a}{v_i}\right]^j_{\underline{v}=\underline{0}}, \left[\frac{\partial \underline{\lambda}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{0}}, \left[\frac{\partial \underline{u}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{0}}, \text{ and } \left[\frac{\partial t_j}{\partial v_i}\right]_{\underline{v}=\underline{0}} \middle|\ j = 1, \ldots, m \right\} \tag{32}$$

where the superscript j on a function of time refers to that function on the time interval $(t_{j-1}(\underline{v}), t_j(\underline{v})]$.

In order to find

$$\left\{ \left[\frac{\partial}{\partial v_i} \underline{w}_a(t, \underline{v})\right]^j_{\underline{v}=\underline{0}} \middle|\ j = 1, \ldots, m \right\},$$

one uses Equations (2) and (11) to get

$$\left[\frac{\partial \underline{w}_a}{\partial v_i}\right]^j_{\underline{v}=\underline{0}} = D(t) \left[\left(\frac{\partial C}{\partial v_i}\right)_{\underline{v}=\underline{0}} \underline{x}^j_a(t, \underline{0}) + C(\underline{0}) \left(\frac{\partial \underline{x}_a}{\partial v_i}\right)^j_{\underline{v}=\underline{0}} + \left(\frac{\partial F}{\partial v_i}\right)_{\underline{v}=\underline{0}} \right] \tag{33}$$

for $t_{j-1}(\underline{v}) < t \le t_j(\underline{v})$, j=1, ..., m.

For each integer j between 1 and m, the above solutions can be substituted into Equations (14), (22), and (23) to give

$$\left\{ S^j(t),\ G^j(t),\ K^j(t) \ \middle|\ j = 1, \ldots, m \right\}.$$

Observe that in order to realize the control defined by Equations (13), (19), (20) and (21) the matrix time functions

$$\left\{ \underline{\lambda}^j_a(t, \underline{0}),\ \underline{x}^j_a(t, \underline{0}),\ \underline{y}^j_a(t, \underline{0}), \right.$$

$$\left. S^j(t),\ G^j(t),\ K^j(t) \ \middle|\ j = 1, \ldots, m \right\}$$

can be computed before the system is used and stored in the computer for the system. Then, when the system is turned on at $t = 0$, the nearly optimal control is immediately generated by Equations (13), (19), (20) and (21) for $t > 0$.

IV EXAMPLE

Suppose
$$\dot{x}_i = x_2,\ x_1(0) = -(1 + v_1),\ x_1(t_f) = 0 \tag{34}$$

$$\dot{x}_2 = (1 + v_3)\, u,\ x_2(0) = -v_2,\ x_2(t_f) = 0 \tag{35}$$

$$\dot{x}_3 = v_2,\ x_3(0) = 1 + v_1,\ x_3(t_f) = \text{free} \tag{36}$$

$$\underline{y} = \underline{x},\ |u| \le 1.0,\ J = t_f \tag{37}$$

Then

$$H = -1 + \lambda_1 x_2 + \lambda_2 (1 + v_3)\, u + \lambda_3 v_2 \tag{38}$$

$$u = \text{sign}\left[(1 + v_3)\lambda_2\right],\ \dot{\lambda}_1 = 0,\ \dot{\lambda}_2 = -\lambda_1,\ \dot{\lambda}_3 = 0 \tag{39}$$

Because of (39), λ_2 changes sign at most once for $0 \le t \le t_f$, u has only one discontinuity for $0 \le t \le t_f$ at time t_1, and $T(\underline{v}) = \{t_1\}$.

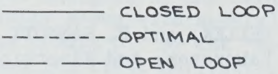
The function $\underline{w}$ is set equal to $\underline{y}$; i.e., $D(t) = I$. Hence, $\underline{w}_a(t, \underline{v}) = [x_{a1}, x_{a2}, x_{a3}]^T$.

The final time $t_{f_c}(t, \underline{y})$ is defined as the time at which $x_2 = 0$. Since the control u_c is an approximation of the optimal feedback solution, in general, $x_1(t_{f_c}) \neq 0$ for every $\underline{v}$. Figures 1, 3 and 5 contain graphs of t_{f_c} as the components of $\underline{v}$ are varied from zero one at a time. Figures 2, 4 and 6 show the corresponding graphs of $x_1(t_{f_c})$. For the purpose of comparison, these figures also contain the corresponding graphs for the optimal solution and for the open-loop nominal solution. Observe from the graphs that the feedback solution approximates the optimal solution much better than the open-loop solution does.

(1) L. S. Pontryagin, The Mathematical Theory of Optimal Processes, Interscience Publishers, 1962, Chapters 1, 2 and 4.

(2) R. A. Werner, "Suboptimal Feedback Control of Systems with Random Parameters," Coordinated Science Laboratory Report R-323, CSL, University of Illinois, Urbana, Illinois, September 1966.

(3) J. V. Breakwell, J. L. Speyer, and A. E. Bryson, "Optimization and Control of Nonlinear Systems Using the Second Variation," Journal on Control, Siam, 1963, pp 193-223.


CLOSED LOOP
OPTIMAL
OPEN LOOP

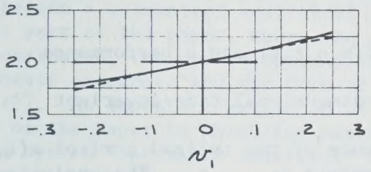


FIGURE 1

t_f VS v_1 WITH $v_2 = 0 = v_3$

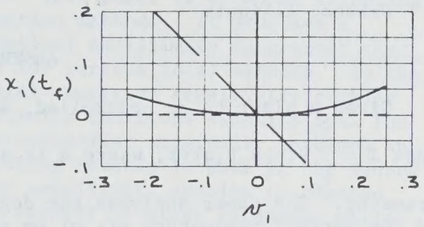


FIGURE 2

$x_1(t_f)$ VS v_1 WITH $v_2 = 0 = v_3$

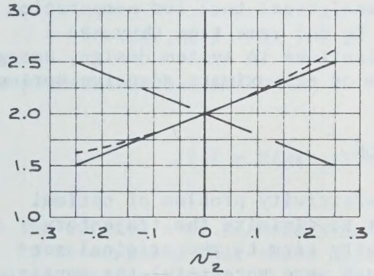


FIGURE 3

t_f VS v_2 WITH $v_1 = 0 = v_3$

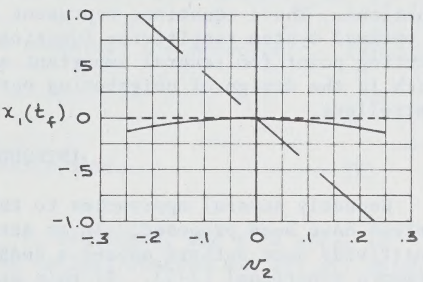


FIGURE 4

$x_1(t_f)$ VS v_2 WITH $v_1 = 0 = v_3$

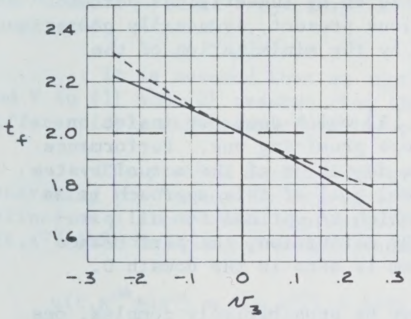


FIGURE 5

t_f VS v_3 WITH $v_1 = 0 = v_2$

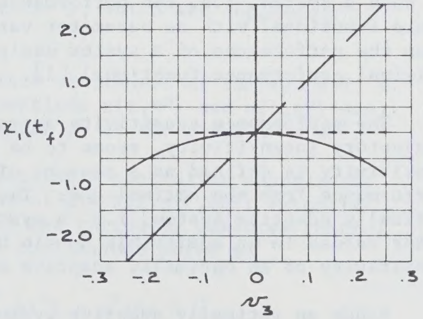


FIGURE 6

$x_1(t_f)$ VS v_3 WITH $v_1 = 0 = v_2$

OBTAINING AND APPLICATIONS OF OPTIMAL SENSITIVITY FUNCTIONS

P. KOKOTOVIĆ,* J. E. HELLER, and P. SANNUTI
Coordinated Science Laboratory and Electrical
Engineering Department, University of Illinois
Urbana, Illinois

ABSTRACT

Given a plant to be controlled, $\dot{x} = f(x,u,t,q)$ and a performance index $J = \int_{t_o}^{t_1} V(x,u,t,q)dt$, where q is an r-dimensional time-invariant parameter. The paper analyzes the dependence of the optimal control u(t,q) and the optimal trajectory x(t,q) on parameters $q_1,\ldots,q_r$. The analysis is based upon the trajectory and control sensitivity matrices $\xi = \{\partial x_i/\partial q_j\}$, $\nu = \{\partial u_k/\partial q_j\}$. The two-point boundary value problem for canonic sensitivity equations is reduced to the solution of one nonlinear matrix differential (Riccati) equation and two linear matrix differential equations. These equations represent an analytical tool for computation of optimal system sensitivity functions. In the same time they are a starting point for several important applications in system design, among which is the design of neighboring optimum or approximate adaptive optimum controllers.

INTRODUCTION

Recently several approaches to the sensitivity problem of optimal systems have been proposed. In an attempt to minimize the trajectory sensitivity some authors append a sensitivity term to the original performance functional [1-5]. If this approach were successful the sensitivity problem would be simply reducible to the minimization of an augmented functional. Unfortunately, the resulting system is insensitive not only to unfavorable parameter variations, but also to those which could have improved its performance. Moreover, the nominal performance of such a system, i.e. the performance measured by the original performance functional with no parameter variations present, is usually poorer than the performance of a system designed by the minimization of the original performance functional [1].

The performance sensitivity approach [6] which does not insist on trajectory insensitivity, seems to be a more promising one. Performance sensitivity is defined as a measure of the deviation of the actual system performance from the optimal one. The ideal goal of this approach is an optimally adaptive system, i.e. a system which is optimal for all parameter values in an admissible domain D. By definition, the performance sensitivity of an optimally adaptive system is zero in the domain D.

Since an optimally adaptive system may be prohibitively complex, one is forced to seek some approximate realizations. Approximation methods proposed in recent papers can be divided into two groups: global methods, usually minimaximizing the performance sensitivity over the whole control and parameter admissible domains [7,8] or using some self-adjusting controllers [9,10]; and local methods approximating the optimally adaptive control at one [11-15] or several [7-16] points of the domain D. Local methods have their origin in the Kelly-Breakwell-Bryson procedure [11,12] and employ truncated series expansions of optimally adaptive control

---

(*) On leave from M.Pupin Research Institute, Belgrade, Yugoslavia

[13-15]. A feedback controller designed by the method developed in [14,15] makes the performance sensitivity along with several of its derivatives zero at a nominal point of the domain D. By the "multipoint design" [7,16] the performance sensitivity and its first derivative are made zero at n+1 "representative" points of the domain D. This paper is aimed at extending and simplifying existing local approximation methods. It proposes a procedure for obtaining and applying "optimal sensitivity functions" that can become a convenient analytical tool for various local methods. In the first part of the paper, where the initial condition sensitivity problem is considered, the use of optimal sensitivity functions simplify both the synthesis procedure and the form of the controller originally obtained in [11,12]. An important extension of existing methods is made in the second part of the paper in which the parameter sensitivity problem is treated and a system "optimally sensitive" with respect to plant parameter variations is synthesized.

OPTIMAL SENSITIVITY WITH RESPECT TO INITIAL CONDITIONS

Given a plant

$$\dot{x} = f(x,u,t)\,;\; x(t_o) = x^o\,;\; x = \text{n-vector};\; u = \text{m-vector}; \qquad (1)$$

and a performance index

$$J(u) = K(x(t_f)) + \int_{t_o}^{t_f} V(x,u,t)\,dt, \qquad (2)$$

where the control u is unconstrained; t_o, t_f are fixed, and $x^f = x(t_f)$ is free. It is supposed that there exists an optimal control $u = u(t,x^o)$ which minimizes the functional (2) for all x^o in a convex domain D. The control $u(t,x^o)$ and the corresponding optimal trajectory $x = x(t,x^o)$ are called "optimally adaptive." The explicit analytic form of the functions $u(t,x^o)$ and $x(t,x^o)$ are unknown. For a "nominal" parameter value $x^o = x^{o*}$ these functions become the "nominal optimal control" $u^* = u(t,x^{o*})$, and the "nominal optimal trajectory" $x^* = x(t,x^{o*})$, which are supposedly numerically obtainable.

It is assumed that an appropriate smoothness of functions f, K and V in (1) and (2) assures that the functions $x(t,x^o)$ and $u(t,x^o)$ are differentiable with respect to the parameter x^o at the point $x^o = x^{o*}$.

The problem consists in synthesizing a controller capable of generating the "optimally sensitive" control $\tilde{u} = \tilde{u}(t,x^{o*},\Delta x^o)$ which is defined as a first order approximation of the optimally adaptive control $u(t,x^{o*}+\Delta x^o)$ in some Δx^o-neighborhood of the nominal point x^{o*},

$$u(t,x^{o*}+\Delta x^o) \approx u(t,x^{o*}) + \delta u(t,x^{o*},\Delta x^o) = u^*+\delta u \equiv \tilde{u}. \qquad (3)$$

Let us introduce an n x n optimal state sensitivity matrix ξ and an mxn optimal control sensitivity matrix ν, whose elements are the optimal sensitivity functions [9,13],

$$\xi^i_j = \xi^i_j(t) = \left.\frac{\partial x_i(t,x^o)}{\partial x^o_j}\right|_{x^o=x^{o*}}, \quad i,j = 1,\dots,n, \qquad (4)$$

$$\nu_j^k = \nu_j^k(t) = \left.\frac{\partial u_k(t,x^o)}{\partial x_j^o}\right|_{x^o=x^{o*}}, \quad k = 1,\ldots,m, \tag{5}$$

respectively. Using ν and ξ, the first variation $\delta u = \tilde{u} - u^*$ and the corresponding trajectory variation $\delta x = x(t,x^{o*} + \Delta x^o) - x(t,x^{o*})$ can be related to Δx^o as follows:

$$\delta u = \nu\,\Delta x^o; \quad \delta x = \xi\,\Delta x^o. \tag{6}$$

Eliminating Δx^o from (6) one obtains the general analytic form of a feedback controller:

$$\delta u = \nu\,\xi^{-1}\,\delta x. \tag{7}$$

For the existence of such a controller it is sufficient that the sensitivity matrices ν and ξ exist and that the state sensitivity matrix ξ is nonsingular for all t, $t \in [t_o, t_f]$. If u^* and x^* are known functions of time, the scheme in Fig. 1 will represent the system defined by relations (3) and (7). As a first order approximation of an optimally adaptive system at $x^o = x^*$, the system in Fig. 1 is "optimally sensitive" with respect to variations of initial conditions x^o.

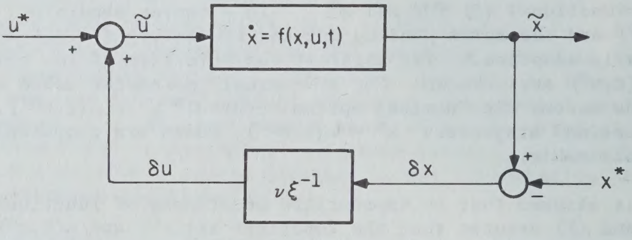


Fig. 1

The conceptual clarity of the formula (7) demonstrates the usefulness of the notion of an optimal sensitivity function. This important formula follows directly from the definition of matrices ν and ξ, and provides a simple and yet exhaustive basis for the synthesis of optimally sensitive systems.

For the remaining part of the synthesis procedure the "optimal sensitivity equations" will be derived from the conditions of the maximum principle.

Let H be the Hamiltonian function for the plant (1) and the performance index (2),

$$H = -V + p'f, \tag{8}$$

where p is the costate variable and the prime denotes transposition. For all $x^o \in D$ and all $t \in [t_o, t_f]$ an optimally adaptive control $u = u(t,x^o)$, and corresponding functions $x = x(t,x^o)$ and $p = p(t,x^o)$ represent the solution of a system of equations,

$$\dot{x} = \nabla_p H, \tag{9}$$

$$\dot{p} = -\nabla_x H, \tag{10}$$

$$0 = \nabla_u H, \tag{11}$$

satisfying the boundary condition

$$x(t_o) = x^o, \tag{9^o}$$

$$p(t_f) = p^f = -\nabla_x K\Big|_{t=t_f}, \tag{10^f}$$

and the condition that the matrix

$$H_{uu} = \frac{\partial(\nabla_u H)}{\partial u} = \frac{\partial^2 H}{\partial u^2} \quad \text{must be negative definite.} \tag{12}$$

We introduce the sensitivity matrix η, whose elements are the costate sensitivity functions

$$\eta_j^r = \eta_j^r(t) = \frac{\partial p_r(t,x^o)}{\partial x_j^o}\Bigg|_{x^o=x^{o*}},\ r,j = 1,\ldots,n, \tag{13}$$

and we assume that, at the point $x^o=x^{o*}$, the conditions are satisfied under which the solution of (9), (10), and (11) has continuous partial derivatives (4), (5), and (13), and under which the mixed partial derivatives $\dot{\xi}_j^i$ and $\dot{\eta}_j^r$ are also continuous and do not depend on the order of differentiation,

$$\dot{\eta}_j^r = \frac{\partial^2 p_r(t,x^o)}{\partial t\, \partial x_j^o} = \frac{\partial^2 p_r(t,x^o)}{\partial x_j^o\, \partial t},$$

$$\dot{\xi}_j^i = \frac{\partial^2 x_i(t,x^o)}{\partial t\, \partial x_j^o} = \frac{\partial^2 x_i(t,x^o)}{\partial x_j^o\, \partial t}.$$

Then, if we differentiate both sides of (9), (10), and (11) with respect to x^o,

$$\dot{\xi} = \frac{\partial(\nabla_p H)}{\partial x}\xi + \frac{\partial(\nabla_p H)}{\partial p}\eta + \frac{\partial(\nabla_p H)}{\partial u}\nu,$$

$$\dot{\eta} = -\frac{\partial(\nabla_x H)}{\partial x}\xi - \frac{\partial(\nabla_x H)}{\partial p}\eta - \frac{\partial(\nabla_x H)}{\partial u}\nu,$$

$$0 = \frac{\partial(\nabla_u H)}{\partial x}\,\xi + \frac{\partial(\nabla_u H)}{\partial p}\,\eta + \frac{\partial(\nabla_u H)}{\partial u}\,\nu$$

and if we notice that

$$\frac{\partial(\nabla_p H)}{\partial x} = \frac{\partial f}{\partial x} = f_x ; \quad \frac{\partial(\nabla_p H)}{\partial p} = 0 ; \quad \frac{\partial(\nabla_p H)}{\partial u} = \frac{\partial f}{\partial u} = f_u ; \quad \frac{\partial(\nabla_x H)}{\partial x} = \frac{\partial^2 H}{\partial x^2} = H_{xx} ;$$

$$H_{xu} = \frac{\partial(\nabla_x H)}{\partial u} = \frac{\partial^2 H}{\partial x \partial u} = H'_{ux} ; \quad \frac{\partial(\nabla_u H)}{\partial u} = \frac{\partial^2 H}{\partial u^2} = H_{uu} ;$$

$$\frac{\partial(\nabla_x H)}{\partial p} = \frac{\partial(f'_x p)}{\partial p} = f'_x ; \quad \frac{\partial(\nabla_u H)}{\partial p} = \frac{\partial(f'_u p)}{\partial p} = f'_u ;$$

we shall obtain the system of linear matrix equations

$$\dot{\xi} = f_x \xi + f_u \nu , \tag{14}$$

$$\dot{\eta} = -H_{xx} \xi - f'_x \eta - H'_{ux} \nu , \tag{15}$$

$$0 = H_{ux} \xi + f'_u \eta + H_{uu} \nu . \tag{16}$$

In view of the definitions (4), (5), and (13) the "nominal optimal functions" u^*, x^*, and p^* have to be employed for the evaluation of matrices f_x, f_u, H_{xx}, H_{ux}, H_{uu}. By virtue of (12) the matrix H_{uu} is non-singular and ν can be expressed from (16)

$$\nu = -H_{uu}^{-1}(H_{ux}\xi + f'_u \eta) , \tag{17}$$

and substituted in (14) and (15),

$$\dot{\xi} = F\xi + S\eta , \tag{18}$$

$$\dot{\eta} = Q\xi - F'\eta , \tag{19}$$

where

$$F = f_x - f_u H_{uu}^{-1} H_{ux} , \tag{20}$$

$$S = -f_u H_{uu}^{-1} f'_u , \tag{21}$$

$$Q = H'_{ux} H_{uu}^{-1} H_{ux} - H_{xx} . \tag{22}$$

It follows from (4) and (9^o) that the initial value of ξ is the n x n identity matrix,

$$\xi(t_o) = I_n , \tag{23}$$

while from (13) and (10^f) one gets

$$\eta(t_f) = \frac{\partial p^f}{\partial x^o} = -\frac{\partial^2 K}{\partial x^2}\,\frac{\partial x^f}{\partial x^o} = -K^f_{xx}\,\xi(t_f). \qquad (24)$$

where K^f_{xx} is the second derivative of K evaluated at $t=t_f$. Linear equations (18) and (19) with the condition (23), (24) represent well known two-point boundary value problem, reducible to the Cauchy problem for a matrix Riccati equation [17]. From (26), (27) it can be concluded that ξ and η are linearly related

$$\eta = -P\xi, \qquad (25)$$

where P=P(t) is an n x n matrix. Differentiating (25) with respect to time,

$$\dot{\eta} = -\dot{P}\xi - P\dot{\xi}, \qquad (26)$$

and eliminating ξ, $\dot{\xi}$, η, and $\dot{\eta}$ from (18), (19), (25), and (26), one obtains a matrix Riccati differential equation for P(t),

$$\dot{P} = -PF - F'P + RSP - Q. \qquad (27)$$

In view of (24) and (25) the matrix P(t) is the solution of (27) satisfying the end condition

$$P(t_f) = -K^f_{xx}. \qquad (27^f)$$

The matrices S and Q, defined by (21) and (22), as well as the matrix K_{xx}, are symmetric, and it is easy to see from (27) that P(t) must be symmetric, too.

Once the matrix P(t) is found, the substitution of (17) and (25) into (7) gives the detailed form of the feedback controller in Fig. 1,

$$\delta u = H^{-1}_{uu}(f'_u P - H_{ux})\,\delta x\ . \qquad (28)$$

The final block-diagram of a system, optimally sensitive with respect to initial condition, is given in Fig. 2. For the synthesis of the controller

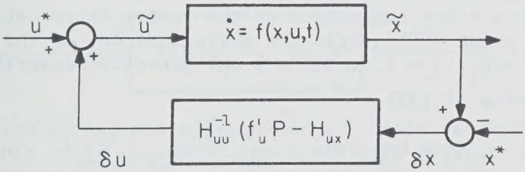


Fig. 2

(28), only the equation (27) has to be solved. If, for the purpose of an analysis, one needs the sensitivity matrix $\xi(t)$, one must also solve the "optimal sensitivity equation"

$$\dot{\xi} = (F-SP)\xi;\ \xi(t_o) = I_n, \tag{29}$$

which is obtained by elimination of η from (18) and (25). With P(t) and $\xi(t)$ known, the sensitivity ν can be computed from (7) and (28).

OPTIMAL SENSITIVITY WITH RESPECT TO PLANT PARAMETERS

The results from the first part of the paper will now be extended to the more complicated problem of optimal sensitivity with respect to an r-dimensional parameter q of an s-dimensional plant

$$\dot{y} = g(y,u,t,q);\ y(t_o) = y^o, \tag{30}$$

where again the m-dimensional control u is unconstrained and the function g has the usual differentiability properties.

By introducing an n-dimensional "super-state" x and an n-dimensional function f,

$$x = \begin{bmatrix} y_1 \\ \vdots \\ y_s \\ q_1 \\ \vdots \\ q_r \end{bmatrix};\quad f = \begin{bmatrix} g_1 \\ \vdots \\ g_s \\ 0 \\ \vdots \\ 0 \end{bmatrix};\quad n = s+r; \tag{31}$$

we rewrite the plant equation (3) in the form (1). Then we have

$$x_{s+j}(t) = x_{s+j}(t_o) = x^o_{s+j} = q_j;\ j = 1,\dots,r; \tag{32}$$

and the parameter sensitivity problem for the plant (30) can be stated as the initial condition sensitivity problem for a plant of the type (1) and the performance index (2). The two problems are equivalent in the sense that for both of them an optimally sensitive controller is given by (28), where the matrix P is the solution of (27). There is, however, a reason why the parameter sensitivity problem is a more difficult one. The reason is that only s = n-r components of the vector δx are obtainable from the state $\tilde{y}$ of the plant (30), while the variations Δq_j of the remaining r coordinates $x_{s+j} = q_j$; j = 1,...,r; are nor directly measurable. If we notice that, by virtue of (31),

$$f'_u = [g'_u \,\vdots\, 0];\quad H_{ux} = [H_{uy} \,\vdots\, H_{uq}]; \tag{33}$$

and if we partition the symmetric nxn matrix P,

$$P = \left[\begin{array}{c|c} P_1 & P_2 \\ \hline P_3 & P_4 \end{array}\right];\ P_3 = P'_2; \tag{34}$$

where P_1, P_2, and P_4 are sxs, sxr, and rxr matrices, respectively, we get from (28):

$$\delta u = H_{uu}^{-1}(g_u' P_1 - H_{uy})\delta y + H_{uu}^{-1}(g_u' P_2 - H_{uq})\Delta q. \qquad (35)$$

The partitioning of all matrices appearing in (27) gives the equations for P_1 and P_2:

$$\dot{P}_1 = -P_1 F_1 - F_1' P_1 + P_1 S_1 P_1 - Q_1, \qquad (36)$$

$$\dot{P}_2 = -P_1 F_2 - F_1' P_2 + P_1 S_1 P_2 - Q_2, \qquad (37)$$

where the matrices F_1, F_2, S_1, Q_1, Q_2, obtained from (20), (21), (22), and (31), are:

$$F_1 = g_y - g_u H_{uu}^{-1} H_{uy}; \; F_2 = g_q - g_u H_{uu}^{-1} H_{uq}; \qquad (38)$$

$$S_1 = -g_u H_{uu}^{-1} g_u' ;$$

$$Q_1 = H_{yu} H_{uu}^{-1} H_{uy} - H_{yy}; \; Q_2 = H_{yu} H_{uu}^{-1} - H_{yq}. \qquad (40)$$

Evidently, the Riccati equation (36) corresponds to initial condition sensitivity problem for the s-dimensional plant (30). Once (36) is solved for P_1, the matrix P_2 can be obtained from the linear equation (37) that represents the additional relation introduced by the parameter sensitivity problem. The end conditions for (36) and (37) are easily seen from (27^f). With $P_1(t)$ and $P_2(t)$ known, the expression (35) defines a controller that makes the system in Fig. 3 optimally sensitive with respect to both initial condition y^o and plant parameter q. Since δu depends on Δq, (35), the system in Fig. 3 is optimally sensitive with respect to q only if Δq is known or obtainable from some measurements of $\tilde{y}$ or δy. We shall demonstrate that optimal sensitivity functions provide a tool for solving this "inverse sensitivity problem"[18].

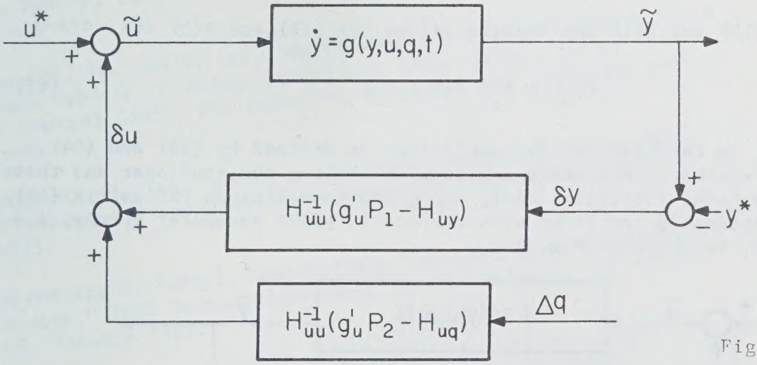


Fig. 3

As a consequence of (31) and (32) the optimal sensitivity matrix ξ of the "super state" x gets the form

$$\xi = \left[\begin{array}{c|c} \zeta & \sigma \\ \hline 0 & I_r \end{array}\right], \tag{41}$$

where ζ and σ are optimal initial condition sensitivity matrix and optimal plant parameter sensitivity matrix of the state y,

$$\zeta^i_j = \frac{\partial y_i}{\partial y^o_j}, \quad \sigma^i_k = \frac{\partial y_i}{\partial q_k}; \quad \begin{array}{l} i,j = 1,\dots,s; \\ k = 1,\dots,r; \end{array} \tag{42}$$

and I_r is the rxr identity matrix. Equation $\delta x = \xi \Delta x^o$ reduces to

$$\delta y = \zeta \Delta y^o + \sigma\, \Delta q. \tag{43}$$

Let us first suppose that r=s and that the inverse σ^{-1} exists. Then a parameter variation Δq can be expressed explicitly

$$\Delta q = \sigma^{-1}(\delta y - \zeta\, \Delta y^o), \tag{44}$$

and, hence, the next step in the synthesis of the controller in Fig. 3 is the computation of matrices σ and ζ. In view of (34) and (38-41) the optimal sensitivity equation (29) splits into two equations:

$$\dot{\zeta} = (F_1 - S_1 P_1)\zeta, \tag{45}$$

$$\dot{\sigma} = (F_1 - S_1 P_1)\sigma + (F_2 - S_1 P_2), \tag{46}$$

owing to (23) and (41) the initial values for $\zeta(t)$ and $\sigma(t)$ are

$$\zeta(t_o) = I_s, \ \sigma(t_o) = 0. \tag{47}$$

Therefore, in the case r=s the controller is defined by (35) and (44), while the synthesis procedure consists in solving one nonlinear and three linear equations (36),(37), (45), (46), and inverting an rxr matrix (44). A system optimally sensitive with respect to plant parameter g only, i.e. when $\Delta y^o = 0$, is shown in Fig. 4.

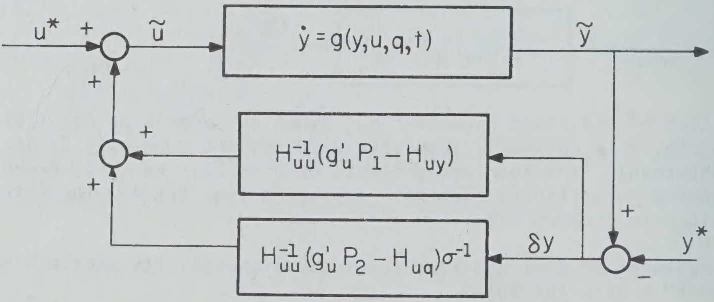

u^*
+
$\tilde{u}$
$\dot{y} = g(y,u,q,t)$
$\tilde{y}$
+
+
+
$H_{uu}^{-1}(g'_u P_1 - H_{uy})$
$H_{uu}^{-1}(g'_u P_2 - H_{uq})\sigma^{-1}$
δy
+
y^*
−

Fig. 4

In the case when $r \neq s$ the term σ^{-1} could denote the generalized inverse of the matrix σ [19,20]. Thus the above discussion carries over to the general parameter sensitivity problem.

It is apparent from the above results that the synthesis of a system optimally sensitive with respect to plant parameter variations is more difficult than the synthesis of a system optimally sensitive with respect to initial condition variations. The parameter sensitivity problem is reducible to the initial condition formulation only as far as the analysis is concerned. For the synthesis of the controller in Fig. 2 one need not invert ξ since in equation (28) it has been canceled out. The basic difficulty in the parameter sensitivity synthesis (Figs. 3, 4) lies in the fact that identification of parameter variation Δq is unavoidable and must be considered either in the form of an inverse σ^{-1} or by some parameter estimation schemes. This difference makes parameter sensitivity a new synthesis problem and further investigations should reveal to what extent the identification process should be incorporated in the controller structure.

REFERENCES

1. P. Kokotović, S. Bingulac, and J. Medanić, "Some Approaches to the Reduction of Control System Sensitivity," 3rd Allerton Conference on Circuit and System Theory, (University of Illinois, Urbana) 1965, pp. 212-217.

2. W. G. Tuel, I. Lee, and P. M. DeRusso, "Synthesis of Optimal Control Systems with Sensitivity Constraints," Session 24, Paper 24B, Third Congress of IFAC, London 1966.

3. H. D'Angelo, M. L. Moe, and T. C. Hendricks, "Trajectory Sensitivity of an Optimal Control System," Proc. 4th Allerton Conf. on Circuit and System Theory, (University of Illinois, Urbana), October 5-7, 1966, pp. 489-498.

4. J. L. Cassidy, Jr. and I. Lee, "On the Optimal Feedback Control of a Large Launch Vehicle to Reduce Trajectory Sensitivity," Joint Automatic Control Conf., Philadelphia, Pa. June 1967 (preprint vol., pp. 587-595).

5. H. J. Dougherty, I. Lee, and P. M. DeRusso, "Synthesis of Optimal Feedback Control Systems Subject to Parameter Variations," ibid, pp. 125-133.

6. R. A. Rohrer and M. Sobral, "Sensitivity Considerations in Optimal System Design," IEEE Transactions on Automatic Control, Vol. AC-10, pp. 43-48, January 1965.

7. D. M. Salmon, "Minimax Controller Design," Coordinated Science Laboratory Report R-358, University of Illinois, July 1967.

8. J. Medanić, "Three Segment Method in the Sensitivity Design of Control Systems," Proc. Fifth Allerton Conference on Circuit and System Theory, (University of Illinois, Urbana), October 4-6, 1967

9. P. Kokotović and R. Rutman, "Sensitivity of Automatic Control Systems," (survey), Avtomatika i Telemehanika, No. 4, Moscow, 1965 (in Russian). English translation in Automation and Remote Control, pp. 727-748.

10. P. Kokotović, J. Medanić, M. Vusković, and S. Bingulac, "Sensitivity Method in the Experimental Design of Adaptive Control Systems," Third Congress of IFAC, London, June 1966, pp. 45B1-45B12.

11. J. V. Breakwell and A. E. Bryson, "Neighboring-Optimum Terminal Control for Multivariable Nonlinear Systems," SIAM Symp. on Multivariable System Theory, Cambridge, Mass. 1962. Also J. V. Breakwell, J. L. Speyer, and A. E. Bryson, "Optimization and Control of Nonlinear Systems using the Second Variation," J. SIAM Control, Ser. A, Vol. 1, No. 2,1963.

12. H. J. Kelley, "Guidance Theory and Extremal Fields," IRE Trans. on Automatic Control, October 1962, pp. 75-82.

13. H. J. Kelley, "An Optimal Guidance Approximation Theory," IEEE Trans. on Automatic Control, Vol. AC-9, No. 4, October 1964, pp. 375-380.

14. R. A. Werner, "Suboptimal Feedback Control of Systems With Random Parameters," Coordinated Science Laboratory Report R-323, University of Illinois, September 1966.

15. R. A. Werner and J. B. Cruz, Jr., "Feedback Control which Preserves Optimality for Systems With Unknown Parameters," to be published in IEEE Trans. on Automatic Control.

16. D. M. Salmon and P. Kokotović, "Multi-Point Design of Time-Varying Feedback Controllers," Proc. Fifth Allerton Conference on Circuit and System Theory, (University of Illinois, Urbana), October 4-6, 1967.

17. R. E. Kalman, "Contributions to the Theory of Optimal Control," Bol. Soc. Math, Mexico, pp. 102-119, 1960.

18. S. P. Bingulac, "Minimum Sensitivity Adaptive Systems," Proc. of IFAC/ETAN Symposium on Sensitivity Analysis, Dubrovnik 1964.

19. T. N. E. Grenville, "The Pseudo Inverse of a Rectangular or Singular Matrix and its Applications to the Solution of System of Linear Equations," SIAM Review, Vol. No. 1, January 1959, pp. 38-43.

20. R. Deutsch, "Estimation Theory," Prentice-Hall, Inc. Englewood Cliffs, N.J., 1965.

ACKNOWLEDGMENTS

The authors are thankful to Professors J. B. Cruz, Jr. and W. R. Perkins for permanent stimulating discussions.

The work reported in this paper has been supported by JSEP under Army Contract DAAB-07-67-C-0199 and Air Force Contract AFOSR 931-67.

THREE SEGMENT METHOD IN THE SENSITIVITY DESIGN OF CONTROL SYSTEMS

JURAJ MEDANIĆ*
Coordinated Science Laboratory, University of Illinois
Urbana, Illinois

ABSTRACT

Situations leading to the sensitivity problem involve two sets U and V, where U is the set of admissible controls and V is the set of admissible perturbations (plants, disturbances or initial conditions). A functional J(u,v) serves as a measure of comparison, but the selection of the best, "sensitivity control," from U for the whole set V is not at all obvious. What is needed is an ordering of controls $u \in U$ with respect to J for the whole set V and the results presented in this paper are a consequence of an ordering performed by utilizing the concept of the segment E(u) corresponding to the control u.

INTRODUCTION

Various definitions of the appropriate system sensitivity have been given. In conventional control theory, the sensitivity of the output to plant variations was usually considered. In optimal control theory, the sensitivity of the performance index to plant variations has drawn attention of numerous research workers.[1,2,3,4,5]

As opposed to the synthesis of optimal control for a given plant, the aim of sensitivity design is to obtain a control assuring satisfactory performance of the system for all plants from an admissible domain. In order to compare various admissible controls a segment of efficiency is introduced. The segment gives the range of possible values of the performance index over the whole set of admissible plants if the system is controlled by a given admissible control. For comparison purposes particularly important are the pessimistic segment corresponding to the control optimal for the worst plant and the optimistic segment corresponding to the control optimal for the best plant. In general, neither the pessimistic nor the optimistic segment will be such that the corresponding controls will secure satisfactory performance of the system for all plants in the admissible domain. In the sensitivity design of control systems we seek a realistic segment such that the corresponding control secures satisfactory performance of the control system over the whole set of admissible plants. Hence we call this method the three segment sensitivity design of control systems.

The procedure may be divided into a part in which the analysis of the problem is performed and a part in which the synthesis procedure is proposed. In the analysis of the problem we compare the relative merits of the possible designs: of the worst case--pessimistic design, the best case--optimistic design, as well as other possibilities. In the synthesis a design most satisfactory with respect to the sensitivity of the performance index is selected and the corresponding realistic control obtained.

The paper is divided into five sections. In the first section we define the segment, in particular the pessimistic and the optimistic

---

*On leave from Mihailo Pupin Institute, Belgrade Yugoslavia.

segments. In the second section we deduce some properties of segments and corresponding controls and introduce the notion of the satisfactory segment and the set of satisfactory controls. In the third section we propose a design procedure and introduce the notion of the sensitivity curve. The fourth section contains further ordering of segments and gives a deeper insight into the possibilities of a realistic design. The fifth section contains conclusions as well as indications for further research along proposed lines.

SEGMENTS

For our purposes the optimal control problem can be stated in the following way: given a performance index J and a plant v^o belonging to an admissible set V find the control $\bar{u}$, from an admissible domain U such that

$$J(\bar{u},v^o) = \min_{u \in U} J(u,v^o). \tag{1}$$

The sensitivity problem in optimal control arises because the actual plant v in the system is for various reasons different from the hypothetical plant v^o used in the design. Hence, one of the important questions is the variation of the performance index if a controller, optimal for the plant $v^o \in V$ is used to control the plant $v \in V$. The sensitivity of the performance index to plant variations will give indication as to what extent is the control of the system by that specific nominal controller feasible or should additional steps be undertaken.

Situations leading to the sensitivity problem therefore involve two sets, U and V, where U is the set of admissible controls and V is the set of admissible perturbations (plants, disturbances or initial conditions) and while the functional J(u,v) may still serve as a measure of comparison, the selection of the best, "sensitivity control," from U for the whole V is not at all obvious.

What is needed is an ordering of controls $u \in U$ with respect to J for the whole set V and the results presented in this paper are consequence of an ordering performed by utilizing the concept of the segment E(u) corresponding to the control u.

Let it be assumed that J(u,v) is continuous and defined on the sets U and V. The segment E(u) corresponding to the control u consists of the range of values that the performance functional J can take on over the whole set V if the system is controlled by the control u. The segment is appropriately defined by $E(u) = [J_1, J_h]$ where J_1 and J_h are the lower and upper bounds respectfully:

$$J_1(u) = \min_{v \in V} J(u,v) \tag{2}$$

$$J_h(u) = \max_{v \in V} J(u,v). \tag{3}$$

Of special interest for further analysis will be the pessimistic and the optimistic segment. The pessimistic segment corresponds to the control optimal for the least favorable plant and its upper bound is defined by

$$P_h = J(u^s,v^s) = \min_{u\in U} \max_{v\in V} J(u,v) ; \tag{4}$$

the lower bound of the pessimistic segment is

$$P_1 = \min_{v\in V} J(u^s,v) . \tag{5}$$

The optimistic segment corresponds to the control optimal for the most favorable plant and its lower bound is therefore defined by

$$O_1 = J(u^b,v^b) = \min_{u\in U} \min_{v\in V} J(u,v) \tag{6}$$

the upper bound of the optimistic segment is

$$O_h = \max_{v\in V} J(u^b,v) . \tag{7}$$

PROPERTIES OF SEGMENTS

The controls u^b and u^s are characteristics of the system; the first allows the possibility of obtaining the best achievable value of performance, $O_1=J(u^b,v^b)$, and the second secures a minimal upper bound of the performance index for all $v\in V$, namely $P_h=J(u^s,v^s)$. The segment corresponding to these two controls are fundamental for the analysis of the sensitivity problem and should be obtained for a meaningful sensitivity design. They give a basis for comparison of the relative merits of the optimistic and pessimistic designs and indicate the need for a further, realistic design of the controller. The importance of these two segments becomes more evident if their simple property is noted:

The pessimistic segment is always contained in the optimistic segment.

The proof is straightforward: by definition $J(u^b,v^b) \equiv O_1 \le P_1$; on the other hand $P_h \equiv J(u^s,v^s) \le J(u,v^s)$ and since this must be valid for any $u\in U$ we have $P_h \le J(u^b,v^s)$ and by definition $J(u^b,v^s) \le J(u^b,v^h) \equiv O_h$.

To note the importance of this property in the sensitivity design of control systems consider the following extreme, but possible situations. Let

$$\delta_1 = P_1 - O_1 \tag{8}$$

$$\delta_h = O_h - P_h \tag{9}$$

and let the system and the performance index be such that the following situations occur, as depicted in Fig. 1:

$$\text{(i)} \quad \delta_h \gg \delta_1, \quad E(u^b) \gg E(u^s) \tag{10}$$

$$\text{(ii)} \quad \delta_h \ll \delta_1, \quad E(u^b) \gg E(u^s)$$

$$\text{(iii)} \quad \delta_h \approx \delta_1, \quad E(u^b) \gg E(u^s) .$$

In the first case the pessimistic design is satisfactory since the control u^s does not permit high values of the performance index and on the other hand not much is lost in comparison with the best achievable performance. In the second case the pessimistic design is not satisfactory. The segment corresponding to the control u^s is very narrow and much of the good performance of the system for variations other than the worst is lost due to the pessimistic design of the controller. It is also evident that the optimistic design is the best design since only a small increase of the upper bound, from P_h to O_h, lends an ostensible decrease of the lower bound, from P_1 to O_1. With respect to these two cases the design can be terminated by choosing the pessimistic design in case (i) and the optimistic design in case (ii).

It is because of the more general situation depicted in case (iii) that there exists a need for further - realistic - design of the control system. In this case the shortcomings of the pessimistic design, displayed in the narrow segment $E(u^s)$ must be compared to the shortcomings of the optimistic design displayed in the high value of O_h. Assuming that we are not satisfied with either of these two segments, that is with neither of the two design possibilities we face the problem of obtaining a control u^o which will give more satisfactory performance for all plants in V.

To clarify what may be expected from such a design observe that with respect to their corresponding segments controls $u \in U$ can be classified into four types. Let J_1 and J_h be the lower and upper bound of the segment corresponding to a control u. Then there exist only such controls that

$$\begin{array}{llll} \text{(i)} & O_1 \leq J_1 \leq P_1 & P_h \leq J_h \leq O_h & \\ \text{(ii)} & O_1 \leq J_1 \leq P_1 & J_h \geq O_h & \\ \text{(iii)} & O \quad J_1 \geq P_1 & P_h \leq J_h \leq O_h & (11) \\ \text{(iv)} & J_1 \geq P_1 & J_h \geq O_h \; . & \end{array}$$

Controls of type (ii) are inferior to the optimistic design, controls of type (iii) are inferior to the pessimistic design and those of type (iv) are inferior to both the optimistic and the pseeimistic design. This leaves us with controls belonging to type (i) as the only candidates for a further design of the control system. The set of controls will be denoted by U_o and called the set of satisfactory controls. To each $u \in U_o$ there corresponds a satisfactory segment E(u) such that

$$E(u^s) \subset E(u) \subset E(u^b) \; . \qquad (12)$$

SENSITIVITY CURVE

Consider now a possible design philosophy. The first step of the design would be to obtain the optimistic and the pessimistic segments. In case neither of these two segments is satisfactory there exists the possibility of defining a desired satisfactory segment and obtaining the corresponding satisfactory control $u^o \in U_o$.

The segment will be defined by selecting its end points. In general, it is not possible to choose arbitrarily both end points of the segment as they are not independent. Therefore, the designer states an allowable upper bound value of the satisfactory segment, for example by specifying that it must satisfy the inequality

$$\frac{J_h - P_h}{P_h} \leq \sigma \ . \tag{13}$$

The corresponding control will then be obtained as the solution of

$$\max_{v \in V} J(u^o, v) = J_h = (1+\sigma) P_h . \tag{14}$$

In general there will be a set $U_1 = \{u_1^o, u_2^o, \ldots,\}$ of controls with the same upper bound and as the realistic control u^o we choose the control with the minimal lower bound of the corresponding segment.

Two questions now arise: Does the realistic segment for an arbitrary J_h belong to the set of satisfactory segments and if so, how does one choose an optimal J_h, i.e. an optimal segment? To analyze these questions more thoroughly the notion of the sensitivity curve is introduced. Consider a coordinate system in which the lower bounds of segments are placed on the vertical axis and the upper bound of segments are placed on the horizontal axis. As a result points P and O are obtained. If σ is now used as a parameter determining the value of the upper bound, and if for all $\sigma \in [0, \sigma_{max}]$, where

$$\sigma_{max} = \frac{O_h - P_h}{P_h} \tag{15}$$

the corresponding realistic segment with the minimal lower bound is obtained, a sensitivity curve connecting points P and O will result. The ordinate of any point on the curve is obtained in the following way. For the choosen σ a set of controls satisfying (14) is obtained and denoted as before by U_1. The controls from the set U_1 can be ordered with respect to the value of the lower bound of their segments:

$$J_1(u, v^1) = \min_{v \in V} J(u, v) \quad \text{for all } u \in U_1 \tag{16}$$

(in general a different v^1 for different $u \in U_1$). The realistic control from the set U_1 will be the control with the minimal lower bound of the corresponding segment, and the value of this bound,

$$J_1(u_1, v^1) = \min_{u \in U_1} J_1(u, v^1) \tag{17}$$

defines the ordinate of the point on the sensitivity curve, whose abscisa is $(1+\sigma) P_h$. If the procedure is repeated for all σ in the mentioned interval the sensitivity curve is obtained. The form of the curve is assumed to be arbitrary for now and on Fig. 2 we have three particular cases: (i) a system in which realistic segments belong to the set of satisfactory segments for all σ, (ii) a system in which there is a subinterval of $[0, \sigma_{max}]$ in which the realistic segments do not belong to the set of satisfactory segments and finally, (iii) a special case of (i) in which there exists a particular segment defined for some value σ_1 by $J_1 = L_1$ and $J_h = H_1$ more favorable than neighboring realistic segments.

If the sensitivity curve were known for a given system it would remain for the designer to pick-out the most favorable point on the sensitivity curve and choose the corresponding control as the most satisfactory. However, as the determination of the sensitivity curve would entail a computation of realistic segments for the whole interval $[P_h, O_h]$ we are compelled to try to obtain some general knowledge about the form of the curve.

ORDERING OF SEGMENTS

If realistic segments are to belong to the set of satisfactory segments for all σ, the sensitivity curve must be contained in the shaded portion of the plane (Fig. 2). If there exists a subinterval of $[0, \sigma_{max}]$ for which realistic segments do not belong to the set of satisfactory segments the sensitivity curve must leave the shaded portion, as shown in Fig. 3. However, due to the continuity of J on the sets U and V and the fixed points P and O, the curve must necessarily return to the shaded area before J_h reaches O_h.

To draw the necessary conclusions we introduce an auxiliary sensitivity curve defined in the following way. Consider segments obtained by fixing the value of the lower bound of the segment. Assume that the lower bound has been fixed as J_1 defined by (analogous to (13)):

$$J_1 = (1+\rho)O_1 \tag{18}$$

and that a set of controls U_2 satisfying the condition

$$\min_{v \in V} J(u,v) = (1+\rho)O_1 \tag{19}$$

has been obtained. For all $u \in U_2$ we find the upper bounds of the corresponding segments,

$$J_2(u,v^2) = \max_{v \in V} J(u,v), \quad \text{for all } u \in U_2 \tag{20}$$

and take from this ordered set, as the realistic control, the control $u_2 \in U_2$ for which the upper bound of the corresponding segment is minimal, and the value of this bound

$$J_2(u_2,v^2) = \min_{u \in U_2} J_2(u,v^2) \tag{21}$$

defines the abscisa of the point on the auxiliary sensitivity curve whose ordinate is $(1+\rho)O_1$. If the procedure is repeated for all $\rho \in [0, \rho_{max}]$, where

$$\rho_{max} = \frac{P_1 - O_1}{O_1} \tag{22}$$

the auxiliary sensitivity curve connecting points O and P will result. If in this case there exists a subinterval of $[0, \rho_{max}]$ for which realistic segments do not belong to the set of satisfactory segments the curve will leave the shaded portion as indicated in Fig. 4, but it must return to the shaded portion before reaching the point P.

With the help of these sensitivity curves we are now ready to prove the following theorem:

> Theorem: Let there be given a control system, a set of admissible controls U and a set of perturbations V and a functional $J(u,v)$ defined and continuous on the sets U and V. Let the segments corresponding to the optimistic and the pessimistic design be denoted by $E(u^b)$ and $E(u^s)$ respectfully. Then there exist realistic segments $E(u)$ such that
>
> $$E(u^s) \subset E(u) \subset E(u^b)$$
>
> for all $J_h \in [P_h, O_h]$, if the ordering of realistic segments is performed with respect to the value of the upper bound and for all $J_1 \in [O_1, P_1]$ if the ordering is performed with respect to the value of the lower bound.

This simply means that the sensitivity curve, as well as the auxiliary sensitivity curve must not leave the shaded portion. Accordingly, the proof of the theorem will be based on the form of these two curves although it will be seen that a mathematical formulation of the proof from these considerations is straightforward.

Assume the contrary and consider a given H_o for which we suppose that the ordinate on the first sensitivity curve is out of the shaded portion, necessarily above it (Fig. 4). Because of continuity there exists a point of the second sensitivity curve having the same H_o and being necessarily in the shaded portion since the second sensitivity curve is defined only on $[O_1, P_1]$. This implies that point L_o cannot be a minimal lower bound point since there exists a segment having the same upper bound H_o but a smaller lower bound, namely L_1. Therefore the first sensitivity curve, due to its definition, must be such that $L_o \leq L_1$, or what is the same must coincide or be under the second sensitivity curve, and hence in the shaded area. Since this must be true for all $H \in [P_h, O_h]$ an immediate consequence is that realistic segments, defined by the sensitivity curve, belong to the set of satisfactory segments for the whole range $[P_h, O_h]$; analogous reasoning could be applied to prove that the same applies for the whole range $[O_1, P_1]$ if the ordering of realistic segments is performed with respect to the value of the lower bound.

> Corollary: The set of realistic segments obtained by ordering with respect to the value of the upper bound coincides with the set of realistic segments obtained by ordering with respect to the value of the lower bound. Or, the auxiliary sensitivity curve coincides with the sensitivity curve.

The first sensitivity curve must necessarily be such that an arbitrary point L_o on it is below or coincides with L_1, a point of the auxiliary curve, for the same H_o. However, if we assume it to be below (the dotted line on Fig. 5), then for some L_2 we have a point H_o on the curve I and a point H_2 on the auxiliary curve and it follows that H_2 is not the minimal upper bound point since there exists a segment with the upper bound $H_o < H_2$.

As a consequence, if J_h is selected as the upper bound and J_1 is the lower bound of the corresponding realistic segment then, upon selecting J_1 as the lower bound, the upper bound of the corresponding realistic segment will be J_h

Corollary: The sensitivity curve is a monotonously non-oncreasing function.

The proof of the corollary follows closely the proof of the theorem. Let us assume that, starting from P we come to a stationary point J at which the assumption of monotonity could be violated, Fig. 6. The point J determines a realistic segment $E(u^J) \subset E(u^b)$ and all conditions for a repetition of the reasoning used in establishing the proof of the theorem are satisfied. Therefore, we conclude that on the portion between J and 0 the sensitivity curve must not leave the shaded area, hence the stationary point cannot be a minimum but only an inflection point. As this must be valid whenever the point J on the sensitivity curve it is sufficient to prove the monotonity of the sensitivity curve.

Before we enter into concluding remarks it is appropriate to consider again the synthesis aspects equipped as we are with the knowledge gained from the above considerations.

Although the choice of the allowable upper bound J_h or the desired lower bound J_1 will depend largely on the designer, the system and of course on any a priori knowledge of the tendency of the plant to vary in some specific direction, it appears that a general synthesis criterion would be to choose as the optimal segment the one corresponding to the point on the sensitivity curve nearest the origin, Fig. 7. This criterion finalizes our ordering of segments from the set of admissible segments, represented by the closed and bounded domain, through the set of satisfactory segments, satisfying (12) and represented by the shaded area, the set of realistic segments, represented by the sensitivity curve and whose properties are given by the two corollaries to the optimal segment and the corresponding sensitivity control defined by

$$\min_{u \in U} \sqrt{J_1^2(u) + J_h^2(u)} \quad . \tag{23}$$

Ways of exploiting this criterion are evident but a more detailed development of these ideas is out of the scope of this paper.

CONCLUSION

The proposed three segment sensitivity design procedure is general and relies on the basic assumption that $J(u,v)$ is defined and continuous on the sets U and V and on the existence of optimal controls and extremalizing perturbations with respect to admissible controls in the sense used throughout the paper.

The analysis of the sensitivity problem has been performed by utilizing the concept of the segment corresponding to an admissible control, in particular by utilizing the pessimistic and the optimistic segments. An ordering of segments from the set of admissible segments, through the sets of satisfactory and realistic segments and finally to the optimal segment and the corresponding sensitivity control determined by (23) defines a criterion for a possible sensitivity design of control systems when the analysis of the problem shows it to be necessary.

From an analytical point of view a major difficulty will be the practical utilization of the criterion defined by (23). The problem can be separated into the basic stage in which a realistic segment is determined for a selected J_h and the advanced stage in which the optimal segment and the corresponding sensitivity control are obtained by iteration of the basic stage. The monotonity of the sensitivity curve simplifies the convergence of the iteration process to the optimal segment and for systems where continuity and differentiability may be assumed, the basic stage is a variational problem with additional side constraints.

Simple examples illustrating the concepts defined in the paper tend to confirm the usefulness of the proposed criterion for a sensitivity design of control systems. In particular, the sensitivity control for the much exploited example of Rohrer-Sobral[1] was extremely near the value obtained by their criterion. This question however needs further consideration before conclusive connections between the two designs can be determined.

ACKNOWLEDGMENT

The author wishes to express his appreciation to Professors Cruz, Kokotovic, and Perkins at the Coordinated Science Laboratory for stimulating discussions and helpful ideas.

REFERENCES

1. Rohrer, R. A. and Sobral, M., "Sensitivity Considerations in Optimal Control Systems," IEEE Trans. on Automatic Control, Vol. AC-10, pp. 43-48, January 1965.

2. Dorato, P., "Sensitivity in Optimal Control Systems," IEEE Trans. on Automatic Control, Vol. AC-8, pp. 256-257, July 1963.

3. Tuel, W. G., Lee, I., and DeRusso, P. M., "Synthesis of Optimal Control Systems with Sensitivity Constraints," III Congress IFAC, London 1966.

4. Cruz, J. B., Jr. and Perkins, W. R., "A New Approach to the Sensitivity Problem in Multivariable Feedback System Design," IEEE Trans. on Automatic Control, Vol. AC-9, July 1964.

5. Mesarović, M. D. and Takahara, Y., "Global Sensitivity of Linear Dynamic Systems," JACC 1967.

This work was supported by the Joint Services Electronics Program (U. S. Army, U. S. Navy, and U. S. Air Force) under Contract DAAB-07-67-C-0199.

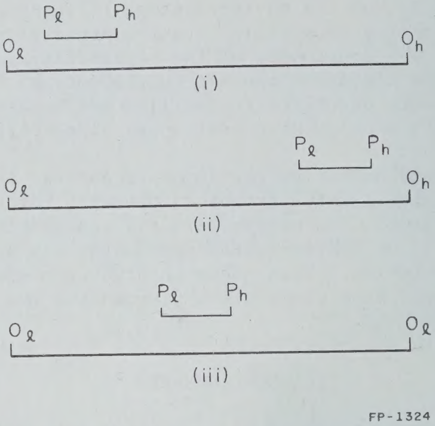

Pℓ
Ph
Oℓ
Oh
(i)
Pℓ
Ph
Oℓ
Oh
(ii)
Pℓ
Ph
Oℓ
Oℓ
(iii)
FP-1324

Fig. 1

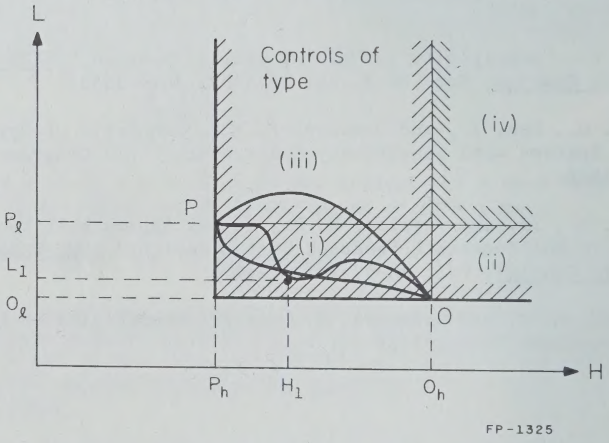
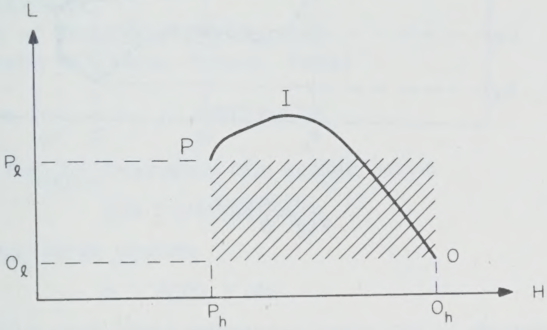
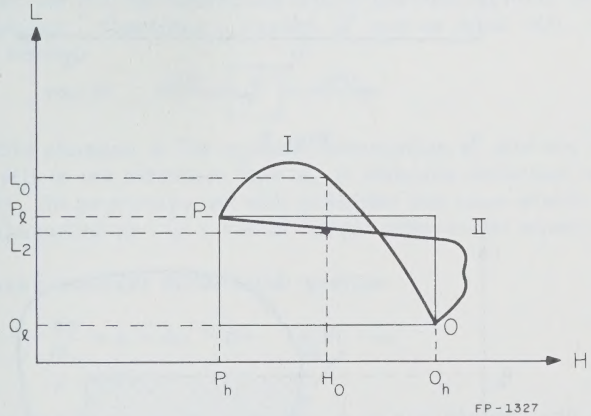

L
Controls of type
(iv)
(iii)
P
Pℓ
(i)
(ii)
L1
Oℓ
O
Ph
H1
Oh
H
FP-1325

Fig. 2



L
I
P
P_ℓ
O_ℓ
O
H
P_h
O_h
FP-1326

Fig. 3



L
I
L_0
P_ℓ
P
II
L_2
O_ℓ
O
H
P_h
H_0
O_h
FP-1327

Fig. 4

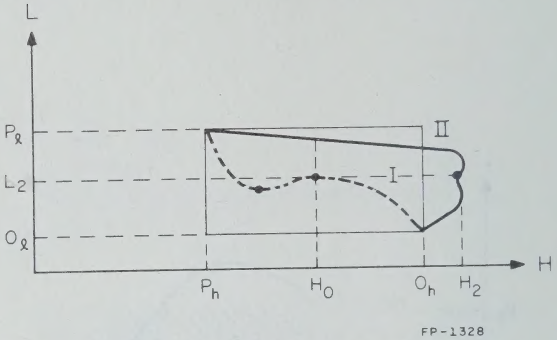

L
P_ℓ
L_2
O_ℓ
II
I
P_h
H_0
O_h
H_2
H
FP-1328

Fig. 5

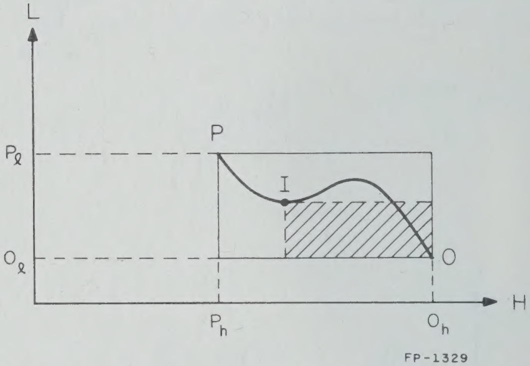

L
P
P_ℓ
I
O_ℓ
O
P_h
O_h
H
FP-1329

Fig. 6

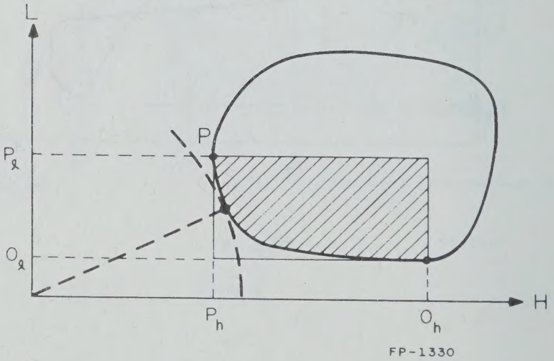

L
P
P_ℓ
O_ℓ
P_h
O_h
H
FP-1330

Fig. 7

QUADRATIC INVARIANCE IN LINEAR SYSTEMS*

N. H. MCCLAMROCH
Information and Control Engineering
The University of Michigan, Ann Arbor, Michigan

J. K. AGGARWAL
Department of Electrical Engineering
The University of Texas, Austin, Texas

ABSTRACT

Conditions for the invariance of a scalar output

$$y(t) = x'(t)\,Q(t)\,x(t)$$

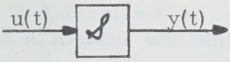
of a time varying linear system

$$\dot{x} = A(t)x + u(t)$$

are obtained. The perturbing influences arise due to changes in the system matrix $A(t)$ and due to additive disturbance inputs which can be represented as changes in $u(t)$. The conditions for invariance of $y(t)$ are obtained when both types of perturbing influences occur simultaneously, and when the perturbing influences do not occur simultaneously. Several observations on the implications of the results are presented.

INTRODUCTION

In this paper we will be concerned with a particular form of the following general problem. Consider a system $\mathscr{S}$ and an input $u(t)$, represented by the diagram below.



What are possible changes in the system description $\mathscr{S}$ and the input $u(t)$ so that the output $y(t)$ is not affected; that is, it remains invariant with respect to these changes. In particular we will consider the case where the state of the system is characterized by a set of linear differential equations.

For the linear constant differential system

$$\frac{dx}{dt} \equiv \dot{x} = Ax + Nv, \quad x(0) = x_0 \tag{1}$$

$$y = Cx \tag{2}$$

where x is an n-vector, v is an r-vector, y is a p-vector, and A, N, and C are constant matrices, Cruz and Perkins [1] have obtained necessary and sufficient conditions for the output y to be invariant with respect to changing A into $A + D$ and with respect to arbitrary inputs $v(t)$. They showed that for the zero input case, $v = 0$, the output is invariant with respect to changing A into $A + D$ if and only if

---

*This research was partially supported by the United States Air Force under Grant No. AF-AFOSR-814-66 and by the National Aeronautics and Space Administration under Grant No. NAS 8-18120.

$$CA^k D = 0, \quad k = 0, 1, \ldots, n-1. \tag{3}$$

They also showed that a necessary and sufficient condition for the output to be invariant with respect to arbitrary inputs v is that

$$CA^k N = 0, \quad k = 0, 1, \ldots, n-1. \tag{4}$$

Further, satisfaction of (3) and (4) is necessary and sufficient for the output to be simultaneously invariant with respect to changing A into A + D and with respect to arbitrary inputs v. Less general results have been obtained by Rutman and Epelman [2] and Rozonoer [3]. It should be noted that conditions (3) and (4) are closely related to the controllability and observability of the system, and that satisfaction of conditions (3) and/or (4) are not likely. Thus we are led to consideration of other types of outputs, with the hope that the conditions for invariance will be less restrictive than conditions (3) and (4).

There have been several investigations into determining the conditions under which an output given by

$$y = \int x' Q x \, dt \tag{5}$$

is invariant with respect to changing A into A + D. Partial results have been obtained by Barnet and Storey [4] and by McClamroch and Aggarwal [5].

The purpose of this paper is to clarify some of the previous results given in [5] and to present some new results associated with the invariance of a quadratic type output. In addition, invariance of the output is considered for time-varying systems with a fixed input u(t).

PRELIMINARIES

In this paper we assume that the system is described by the input-output state relations

$$\dot{x} = A(t)x + u(t), \quad x(t_1) = p \tag{6}$$

$$y = x' Q(t) x \tag{7}$$

where u(t) is the input, specified over the time interval $[t_1, t_2]$, y is the scalar output, and x is an n-vector, the state of the system. A(t) and Q(t) are n × n matrices which are continuous on $[t_1, t_2]$, with Q(t) symmetric. The input u(t) is assumed to be piecewise continuous on $[t_1, t_2]$. The unique solution to (6) is denoted by $x(t, p, t_1)$ or more simply by x(t) when the initial state and initial time are obvious. To indicate the dependence on the initial state and initial time we denote the output (7) by $y(t, p, t_1)$. We are interested in the conditions for which $y(t, p, t_1)$ is invariant with respect to various types of changes in the state description (6).

The following definitions of invariance are similar to the definitions proposed by Cruz and Perkins [1]; however, the definitions presented here apply to a more general class of systems and include the case of finite-time invariance. We consider the cases where changes occur in the matrix A(t) and the vector input u(t), these changes being denoted by E(t) and e(t) respectively. Thus the input-output-state equations become

$$\dot{x} = [A(t) + E(t)]x + u(t) + e(t), \quad x(t_1) = p \tag{8}$$

$$\tilde{y} = x'Q(t)x \tag{9}$$

where $\tilde{y}$ denotes the system output for the state description given by (8). As before $\tilde{y}(t, p, t_1)$ is used to denote the dependence of the output (9) on the initial data. Using this notation we have the following definitions.

Definition 1. The system output is parameter invariant with respect to changing the matrix $A(t)$ into $A(t) + E(t)$ over the time interval $[t_1, t_2]$ if for each initial state $p \in R^n$,

$$y(t, p, t_1) = \tilde{y}(t, p, t_1), \quad t \in [t_1, t_2]$$

with $e(t) \equiv 0$ in (8).

Definition 2. The system output is signal invariant with respect to changing the input $u(t)$ into $u(t) + e(t)$ over the time interval $[t_1, t_2]$ if for each initial state $p \in R^n$,

$$y(t, p, t_1) = \tilde{y}(t, p, t_1), \quad t \in [t_1, t_2]$$

with $E(t) \equiv 0$ in (8).

Definition 3. The system output is simultaneously parameter and signal invariant with respect to changing $A(t)$ into $A(t) + E(t)$ and $u(t)$ into $u(t) + e(t)$ over the time interval $[t_1, t_2]$ if for each initial state $p \in R^n$,

$$y(t, p, t_1) = \tilde{y}(t, p, t_1), \quad t \in [t_1, t_2].$$

In what follows we determine necessary and sufficient conditions that the output be simultaneously parameter and signal invariant. We also determine several conditions for parameter invariance and signal invariance alone. To obtain these conditions we consider the associated output function

$$Y(t, p, t_1) = \int_{t_1}^{t} y(\tau, p, t_1)\,d\tau. \tag{10}$$

To make the proofs of the various results straightforward we first prove two important lemmas. In the first lemma we prove the equivalence of several statements so that we can consider the conditions for invariance of $Y(t_2, p, t)$ rather than $y(t, p, t_1)$.

Lemma 1. The following three statements are equivalent:

(a) $y(t, p, t_1) = \tilde{y}(t, p, t_1)$ for all $p \in R^n$ and all $t \in [t_1, t_2]$.

(b) $Y(t, p, t_1) = \tilde{Y}(t, p, t_1)$ for all $p \in R^n$ and all $t \in [t_1, t_2]$.

(c) $Y(t_2, p, t) = \tilde{Y}(t_2, p, t)$ for all $p \in R^n$ and all $t \in [t_1, t_2]$.

Proof: It suffices to show that (a) and (b) are equivalent and that (b) and (c) are equivalent. It follows directly from (10) and the continuity of $y(t)$ that (a) and (b) are equivalent. We now consider the equivalence of (b) and (c). Statement (b) implies that

$$\int_{t_1}^{t_2} y(\tau, p, t_1)\,d\tau = \int_{t_1}^{t_2} \tilde{y}(\tau, p, t_1)\,d\tau$$

for all $p \in R^n$, and that

$$\int_{t_1}^{t} y(\tau, p, t_1)\,d\tau = \int_{t_1}^{t} \tilde{y}(\tau, p, t_1)\,d\tau$$

for all $p \in R^n$ and all $t \in [t_1, t_2]$. By subtracting the second equation from the first equation, statement (c) is obtained. It is easily shown that statement (c) implies statement (b) by using a similar construction. The proof is completed.

The basic idea of the proof of Theorem 1 in the next section is to make use of the technique of invariant imbedding [6] by considering the initial time and the initial state as independent variables.

Lemma 2. The function $\tilde{Y}(t_2, x, t)$ satisfies the partial differential equation

$$\frac{\partial \tilde{Y}}{\partial t} + \frac{\partial \tilde{Y}'}{\partial x}[A(t) + E(t)]x + \frac{\partial \tilde{Y}'}{\partial x}[u(t) + e(t)] + x'Q(t)x = 0 \quad \text{on } R^n \times [t_1, t_2] \tag{11}$$

and the boundary condition

$$\tilde{Y}(t_2, x, t_2) = 0 \text{ on } R^n. \tag{12}$$

Proof: The Eulerian derivative of $\tilde{Y}(t_2, x, t)$ along the trajectories of (8) is given by

$$\begin{aligned}\frac{d\tilde{Y}}{dt} &= \frac{\partial \tilde{Y}}{\partial t} + \frac{\partial \tilde{Y}'}{\partial x}\dot{x} \\ &= \frac{\partial \tilde{Y}}{\partial t} + \frac{\partial \tilde{Y}'}{\partial x}[A(t) + E(t)]x + \frac{\partial \tilde{Y}'}{\partial x}[u(t) + e(t)].\end{aligned}$$

But from (9) and (10), we see that

$$\frac{d\tilde{Y}}{dt} = -x'Q(t)x,$$

so that (11) is obtained. The boundary condition (12) follows directly from (10), and the lemma is proved.

We have obtained a characterization for $\tilde{Y}(t_2, x, t)$, which denotes the value of the integral (10) along the trajectory of (8) with initial state x and initial time t. It also follows that $Y(t_2, x, t)$ satisfies Lemma 2 with $E(t) \equiv 0$ and $e(t) = 0$ on $[t_1, t_2]$.

RESULTS ON INVARIANCE

In this section we state and prove the results on invariance, making use of Lemmas 1 and 2. We first consider the conditions for the output to be simultaneously parameter invariant and signal invariant.

Theorem 1. A necessary and sufficient condition for the system output to be simultaneously parameter invariant and signal invariant on the time interval $[t_1, t_2]$ is that E(t) and e(t) satisfy

$$P(t)\,E(t) + E'(t)\,P(t) = 0 \quad \text{on } [t_1, t_2] \tag{13}$$

$$2P(t)\,e(t) + E'(t)\,r(t) = 0 \quad \text{on } [t_1, t_2] \tag{14}$$

where P(t) is a symmetric n × n matrix which satisfies

$$\dot{P} + PA + A'P + Q = 0, \quad P(t_2) = 0 \tag{15}$$

and r(t) is an n-vector which satisfies

$$\dot{r} + A'r + 2Pu = 0, \quad r(t_2) = 0 . \tag{16}$$

Proof: We first determine $\widetilde{Y}(t_2, x, t)$ for arbitrary choice of E(t) and e(t) by use of Lemma 2. Lemma 1 is then used to obtain the desired result. Assume that the solution to (11) and (12) is given by

$$\widetilde{Y}(t_2, x, t) = x'\widetilde{P}(t)x + x'\widetilde{r}(t) + \widetilde{s}(t) .$$

Substituting into (11) and (12) we obtain

$$\dot{\widetilde{P}} + \widetilde{P}(A + E) + (A' + E')\widetilde{P} + Q = 0, \quad \widetilde{P}(t_2) = 0$$

$$\dot{\widetilde{r}} + (A' + E')\widetilde{r} + 2\widetilde{P}(u + e) = 0, \quad \widetilde{r}(t_2) = 0$$

$$\dot{\widetilde{s}} + \widetilde{r}'(u + e) = 0, \quad \widetilde{s}(t_2) = 0$$

since (11) and (12) hold for all $x \in R^n$. If E(t) = 0 and e(t) = 0, then we have in a similar fashion that

$$Y(t_2, x, t) = x'P(t)x + r'(t)x + s(t)$$

where

$$\dot{P} + PA + A'P + Q = 0, \quad P(t_2) = 0$$

$$\dot{r} + A'r + 2Pu = 0, \quad r(t_2) = 0$$

$$\dot{s} + r'u = 0, \quad s(t_2) = 0$$

In order that $\widetilde{Y}(t_2, x, t) = Y(t_2, x, t)$ for all $x \in R^n$ and all $t \in [t_1, t_2]$ we require that

$$P(t) = \widetilde{P}(t) \quad \text{on } [t_1, t_2],$$

$$r(t) = \widetilde{r}(t) \quad \text{on } [t_1, t_2],$$

$$s(t) = \widetilde{s}(t) \quad \text{on } [t_1, t_2].$$

It follows from the equations for P, $\widetilde{P}$, r, $\widetilde{r}$, s, $\widetilde{s}$ that these conditions are satisfied if and only if conditions (13) and (14) are satisfied. Thus we have necessary and sufficient conditions that

$$Y(t_2, x, t) = \widetilde{Y}(t_2, x, t) \quad \text{on } R^n \times [t_1, t_2] .$$

But from Lemma 1 this condition is equivalent to the satisfaction of

$$y(t, p, t_1) = \widetilde{y}(t, p, t_1) \quad \text{on } R^n \times [t_1, t_2]$$

which is the desired result.

Theorem 1 holds for arbitrary choice of the output matrix Q(t). We can obtain the following result by placing a restriction on Q(t).

Corollary 1. Assume Q(t) is non-singular on $[t_1, t_2]$. Then the system output is simultaneously parameter invariant and signal invariant if and only if

$$E(t) = 2S(t)P(t) \quad \text{on } [t_1, t_2] \tag{17}$$

$$e(t) = S(t)r(t) \quad \text{on } [t_1, t_2] \tag{18}$$

where P(t) and r(t) satisfy (15) and (16), and S(t) is an arbitrary skew-symmetric matrix,

$$S(t) + S'(t) = 0 \quad \text{on } [t_1, t_2].$$

Proof: The solution for P(t) from (15) can be written as

$$P(t) = \int_t^{t_2} \Phi'(\tau, t)\, Q(\tau)\, \Phi(\tau, t)\, d\tau$$

where $\Phi(\tau, t)$ is the state transition matrix satisfying

$$\frac{d}{d\tau}\Phi(\tau, t) = A(\tau)\,\Phi(\tau, t), \quad \Phi(t, t) = I$$

where I is the $n \times n$ identity matrix. Since $\Phi(\tau, t)$ is never singular, it follows that P(t) is non-singular on $[t_1, t_2)$. Thus the unique solution to (13) and (14) are given by (17) and (18).

In Theorem 1 and Corollary 1 we have obtained necessary and sufficient conditions for the output to be simultaneously parameter invariant and signal invariant. We now consider some conditions for each type of invariance alone. These conditions are easily obtained from the previous results by making the appropriate assumptions about E(t) and e(t). For simplicity we do not consider the most general cases, but rather we present the most obvious results.

Corollary 2. Assume $u(t) = 0$ and Q(t) is non-singular on $[t_1, t_2]$. A necessary and sufficient condition for the system output to be parameter invariant on $[t_1, t_2]$ is that

$$E(t) = S(t)\,P(t) \quad \text{on } [t_1, t_2] \tag{19}$$

for some skew-symmetric matrix S(t), where P(t) satisfies (15).

Corollary 3. Assume Q(t) is non-singular on $[t_1, t_2]$. A necessary and sufficient condition for the system output to be signal invariant on $[t_1, t_2]$ is that

$$e(t) = 0 \quad \text{on } [t_1, t_2] \tag{20}$$

We now consider a slightly different class of systems for which the previous results apply. Assume that the input-state equations are given by (6) or (8) as previously, but the output equation is given by

$$y(t) = \epsilon'(t)\, Q(t)\, \epsilon(t) \tag{21}$$

where $\epsilon(t) = x(t) - q(t)$. Here q(t) is a continuously differentiable vector function defined on $[t_1, t_2]$. The problem now is to determine the conditions so that the output (21) is invariant on the interval $[t_1, t_2]$. These conditions follow directly from the previous results by defining a new set of state equations in the variable ϵ. Corresponding to Theorem 1 and Corollary 1, we obtain the following corollaries.

Corollary 4. A necessary and sufficient condition for the system output (21) to be simultaneously parameter invariant and signal invariant on $[t_1, t_2]$ is that

$$P(t)\,E(t) + E'(t)\,P(t) = 0 \quad \text{on } [t_1, t_2] \tag{22}$$

$$2P(t)\,[e(t) + E(t)\,q(t)] + E'(t)\,r(t) = 0 \text{ on } [t_1, t_2] \tag{23}$$

where P(t) and r(t) satisfy

$$\dot{P} + PA + A'P + Q = 0, \quad P(t_2) = 0 \tag{24}$$

$$\dot{r} + A'r + 2P(u + Aq - \dot{q}) = 0, \quad r(t_2) = 0 . \tag{25}$$

Corollary 5. Assume Q(t) is non-singular on $[t_1, t_2]$. A necessary and sufficient condition for the system output (21) to be simultaneously parameter invariant and signal invariant on $[t_1, t_2]$ is that

$$E(t) = 2S(t)\,P(t) \quad \text{on } [t_1, t_2] \tag{26}$$

$$e(t) = S(t)[r(t) - 2P(t)\,q(t)] \quad \text{on } [t_1, t_2] \tag{27}$$

where P(t) and r(t) satisfy (24) and (25) and S(t) is an arbitrary skew-symmetric matrix on $[t_1, t_2]$.

Additional results, similar to Corollaries 2 and 3, can easily be obtained for the system output (21).

CONCLUSIONS

It is observed by considering the previous results that in general there will be many choices of E(t) and e(t) which satisfy the various conditions for invariance. As an example, consider Corollary 2. The matrix P(t) is fixed, but the skew-symmetric matrix S(t) is completely arbitrary. Thus if the state is n-dimensional, it follows that there can be $n(n - 1)/2$ arbitrary parameters or functions in S, so that E(t) essentially has $n(n - 1)/2$ degrees of freedom which do not affect the system output. Equivalently, we can say that satisfaction of the invariance condition in Corollary 3 imposes $n(n + 1)/2$ constraints on E(t). Similar statements can be made about the other results, although it is not as clear how many arbitrary parameters E(t) and e(t) can include.

As indicated in the Introduction the conditions obtained by Cruz and Perkins for invariance of an output which is a linear combination of the state are rather severe. However, the conditions for invariance of a quadratic output are in general much less restrictive, and they could be satisfied in practice. Thus, by consideration of an output quadratic in the state, rather than linear, we have extended the possible number of cases for which invariance can occur.

In this work we have considered the conditions for invariance of a quadratic type output; but from Lemma 2 it follows that the same conditions apply for the invariance of

$$\int_{t_1}^{t} x'(\sigma)\, Q(\sigma)\, x(\sigma)\, d\sigma$$

for all $t \in [t_1, t_2]$, a functional which has certain interesting connections with optimal control theory. Some of these connections have been considered in [5].

REFERENCES

1. J. B. Cruz and W. R. Perkins, "Conditions for Signal and Parameter Invariance in Dynamical Systems," IEEE Trans. on Automatic Control (Correspondence), vol. AC-11, pp. 614-615, July, 1966.

2. R. S. Rutman and M. S. Epelman, "Parametric Invariance of Linear Dynamic Systems," in Sensitivity Methods in Control Theory, Oxford: Pergammon Press, 1966.

3. L. I. Rozonoer, "A Variational Approach to the Problem of Invariance of Automatic Control Systems," Automation and Remote Control, I, vol. 24, pp. 680-691; II, vol. 24, pp. 793-800.

4. S. Barnet and C. Storey, "Insensitivity of Optimal Linear Control Systems to Persistent Changes in Parameters," Int. J. Control, vol. 4, pp. 41-48, 1966.

5. N. H. McClamroch and J. K. Aggarwal, "On Parameter Invariance in Linear Control Systems," Int. J. Control, (to appear).

6. R. Bellman, R. Kalaba and G. M. Wing, "Invariant Imbedding and Mathematical Physics - I: Particle Processes," J. Math. Phys., vol. 1, pp. 280-308, 1960.

SENSITIVITY IN AN OPTIMAL CONTROL SYSTEM WITH DISTRIBUTED PARAMETERS*

A. J. KOIVUNIEMI P. KRUH
Electrical Engineering
Purdue University
Lafayette, Indiana

The optimal control problem investigated is described by the partial differential equation

$$\frac{\partial^2 q(x,t)}{\partial x^2} = \frac{\partial q(x,t)}{\partial t} \tag{1}$$

It represents a one-dimensional heat conduction. $q(x,t)$ is the temperature distribution on the metal of a furnace heated on one side; $x(0 \leq x \leq 1)$ is the normalized spacial coordinate; t signifies the time. The boundary conditions are specified by

$$\partial q(x,t)/\partial x|_{x=0} = \alpha[q(0,t)-v(t)];\quad \partial q(x,t)/\partial x|_{x=1} = 0;\ \text{and}\ q(x,0)=0,$$

where α is the heat transfer coefficient, and $v(t)$ is the temperature in the gas medium. The temperature of the medium controlled by the fuel flow $u(t)$ evolves in time according to the ordinary differential equation

$$\gamma \frac{dv(t)}{dt} + v(t) = u(t) \tag{2}$$

where γ is the time constant of the furnace. The control is constrained by $0 \leq u(t) \leq 1$ for all t.

The system is optimized relative to the admissible fuel flow by minimizing an index of performance (IP), which is the absolute value of the difference between the desired and the actual temperature distributions evaluated at the terminal time and computed over the range of the spacial coordinate.

The optimum control problem is discretized and posed in the framework of linear programming. The optimal solution is computed by the application of the simplex method of linear programming.

The sensitivity of the optimal solution is investigated relative to system parameters. The parametric sensitivity studied is defined as the partial derivative of the IP with respect to the system parameters evaluated at the nominal values of the parameters. An algorithm is constructed for the determination of the sensitivity of the optimal solution relative to system parameters.

The effect of the values of the system parameters on the optimal value of the IP is studied. A system is determined so that the optimal value of the IP is minimized relative to the parameters. The resulting system is also the least sensitive optimal system. A systematic computational scheme is developed for the design of such a system. The construction of the algorithm is based on the gradient technique in the parametric space. Numerical results obtained from the algorithm are presented; and the convergence of the scheme is discussed.

---

*This research was supported in part by NASA under Contract No. 15-00-021, and in part by NSF Grant No. GK-1970.

ANOTHER EXTENSION OF THE DARLINGTON SYNTHESIS

(T.R.NARASIMHAN AND V.RAMACHANDRAN)

Nova Scotia Technical College, Halifax, Canada.

ABSTRACT

It is shown that Negative Reactive Gyrators (NRG) can be used along with a passive network in the realization of a specified Driving Point Immittance Function (DPI). The technique outlined is an extension of Darlington driving point function synthesis. The realized network is reciprocal using at most two NRGs. The ideal transformers are avoided and the resultant network is a parallel combination of a passive LC network and two NRGs, the entire network being terminated in a one-ohm resistance.

It is also shown that NRGs can be effectively used to realize any specified Y-matrix of the two-element type. The input and output immittances remain invariant and the zeros of transmission can be located anywhere in the s-plane.

(This work is a part of the Ph.D. thesis of T.R.Narasimhan and is supported by the Research and Development Committee of the College.)

DERIVED POSITIVE REAL FUNCTIONS AND MATRICES AND CONSEQUENT COEFFICIENT RELATIONS

N. K. BOSE** AND F. M. REZA*

**Department of Electrical Engineering, University of Pittsburgh

*Department of Electrical Engineering, Syracuse University

ABSTRACT

It was shown previously by one of the authors[1,2] that the positive real property of rational positive real functions representing two-element kind networks, remains invariant under simultaneous differentiation of the numerator and denominator polynomials. Some time back, Talbot[3] proved that the result is valid for any positive real function, and furthermore that it holds for positive but not necessarily real functions.

In this paper, by using an artifice due to Brune it is shown that the invariance of the positive real property under the above-mentioned operation can be extended to rational positive real matrices (open-circuit impedance and short-circuit admittance), whose elements have only common poles.

These results are used to arrive at a set of ordered coefficient inequalities which serve as a set of necessary conditions for two-element kind realizability. A complete chart with the ordered coefficient inequalities, associated with positive real functions representing all possible two-element kind networks, is presented. The chart emphasizes the simplicity of the derived relations. A set of necessary conditions to determine the positive real nature of the type of real rational matrices considered, is expressed in terms of the positive semidefiniteness of certain matrices which are directly obtainable from the coefficients of the original matrix elements.

Notations. To save space, the following abbreviations are used:

$Z(s) = \dfrac{P(s)}{Q(s)}$ is a positive real function.

$Z^n(s) = \dfrac{\dfrac{d^nP(s)}{ds^n}}{\dfrac{d^nQ(s)}{ds^n}}$ n=1,2,..., are the derived positive real functions.

$[\ell_{ij}(s)] = [\dfrac{P_{ij}(s)}{Q_{ij}(s)}]$ = positive real immitance matrix of an n-port network.

$[Z_{ij}(s)]$ = open circuit impedance matrix of an n-port network.

$[y_{ij}(s)]$ = short circuit admittance matrix of an n-port network.

$$[\ell_{ij}^{\ n}(s)] = \left[\frac{\dfrac{d^nP_{ij}(s)}{ds^n}}{\dfrac{d^nQ_{ij}(s)}{ds^n}}\right], \quad n=1,2,\ldots \text{ "derived" positive real matrices.}$$

$\forall_x$ = for all values of the real variable x.

Derived Positive Real Functions

The pertinent part of Talbot's result[3], which will be used in this paper, will be written in the form of a lemma.

Lemma. Let $Z(s) = \frac{P(s)}{Q(s)}$ be a positive real function representing the driving point impedance or admittance of any passive, linear, lumped, finite network. Then $Z^n(s)$ also represents the driving point impedance or admittance of a passive linear, lumped, and finite network.

Using the above result, we will arrive at a similar theorem for positive real matrices.

Derived Positive Real Matrices

Theorem 1. Let $[Z_{ij}(s)] = [\frac{P_{ij}(s)}{Q(s)}]$ be a positive real matrix, whose elements, having only common poles, represent the open-circuit driving-point and transfer impedances of an n-port network. Then, $[Z_{ij}^{\ n}(s)]$ is also a positive real matrix, representing the open circuit driving point and transfer impedances of an n-port network.

Proof:

In Figure 1, as $[Z_{ij}]$ is a positive real matrix,

$$Z(s) = \frac{E(s)}{I(s)} = \sum_{i=1}^{n} \sum_{j=1}^{n} Z_{ij}x_ix_j$$

is a positive real function, for all x_i, x_j, where x_i is the turns ratio of the ideal transformer at the i^{th} port of the n-port network, represented by $[Z_{ij}(s)]$.

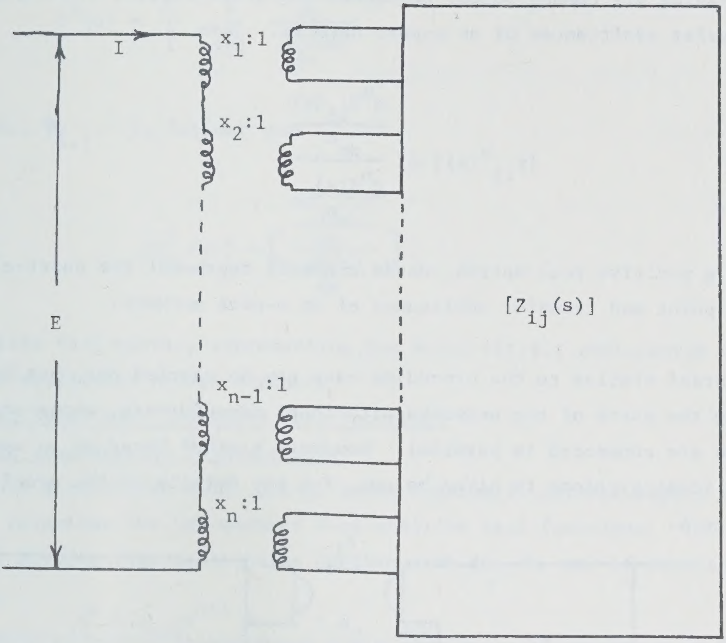


Figure 1

Applying the lemma, it follows that

$$Z^1(s) = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\frac{dP_{ij}(s)}{ds}}{\frac{dQ(s)}{ds}} x_i x_j$$

is a positive real function for all x_i, x_j. Whence

$$[Z^1_{ij}(s)] = \left[\frac{\frac{dP_{ij}(s)}{ds}}{\frac{dQ(s)}{ds}} \right]$$

is a positive real matrix, of rational functions. Such matrices can be realized as the open-circuit impedance matrix of an n-port[4].

Theorem 2. Let $[y_{ij}(s)] = [\frac{R_{ij}(s)}{K(s)}]$ be a positive real matrix, whose elements, having all common poles, represent the short-circuit driving point and transfer admittances of an n-port network. Then

$$[y_{ij}{}^{n}(s)] = \frac{\dfrac{d^{n}R_{ij}(s)}{ds^{n}}}{\dfrac{d^{n}K(s)}{ds^{n}}}$$

is also a positive real matrix, whose elements represent the short-circuit driving point and transfer admittance of an n-port network.

Proof:

A proof similar to the preceding case can be carried out, but by terminating the ports of the networks with ideal transformers, whose primary windings are connected in parallel. However, a proof based on an application of ideal gyrators is given below. For the details of the proof, see Reference 5.

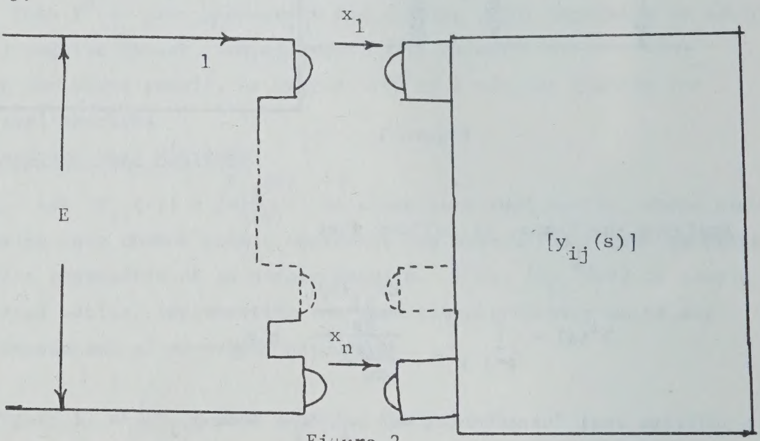


Figure 2

Figure 2 shows an n-port network terminated at each port by a gyrator. It is seen that

$$Z(s) = \frac{E(s)}{I(s)} = \sum_{i=1}^{n} \sum_{j=1}^{n} y_{ij}(s)\, x_i x_j$$

is a positive real function for all x_i, x_j where x_i is the gyration resistance of the gyrator at the i^{th} port.

Using the lemma

$$z^1(s) = \sum_{i}^{n} \sum_{j=1}^{n} \frac{\frac{dR_{ij}(s)}{ds}}{\frac{dK(s)}{ds}}$$

is a p.r.f., $\forall_{i,j}$. It follows that

$$[y^1_{ij}(s)] = \left[\frac{\frac{dR_{ij}(s)}{ds}}{\frac{dK(s)}{ds}} \right]$$

is a positive real matrix, representing the short-circuit admittances of a physically realizable n-port network.

Coefficient Relations. Positive Real Functions

a) RC Impedance or RL Admittance

In this section, we shall drive, using Theorem 1, certain useful coefficient relations for two-element kind positive real functions. The section is divided into three parts dealing with RC, RL, and LC immitance functions.

Let $Z(s) = \dfrac{s^n + a_{n-1}s^{n-1} + \ldots + a_0}{s^n + b_{n-1}s^{n-1} + \ldots + b_0} = \dfrac{P(s)}{Q(s)}$ be an RC impedance or RL admittance driving point function. (The discussion below will be confined to RC impedance functions.)

It is known[1] that $\dfrac{P'(s)}{P(s)}$, and $\dfrac{P'(s)}{Q'(s)}$ are each realizable as RC networks. It is not difficult to show that

$$\frac{P'(\sigma)}{Q'(\sigma)} < \frac{P(\sigma)}{Q(\sigma)} \quad , \quad 0 \leq \sigma < \infty$$

In fact:

$$\frac{dZ(\sigma)}{d\sigma} = \frac{P(\sigma)}{Q(\sigma)} \left[\frac{P'(\sigma)}{P(\sigma)} - \frac{Q'(\sigma)}{Q(\sigma)}\right]$$

As $\dfrac{dZ(\sigma)}{d\sigma} < 0$, and in $0 \leq \sigma < \infty$, $\dfrac{P(\sigma)}{Q(\sigma)} > 0$, the stated result follows. This implied that for $0 \leq \sigma < \infty$, we have

$$\frac{P^n(\sigma)}{Q^n(\sigma)} < \frac{P^{n-1}(\sigma)}{Q^{n-1}(\sigma)} < \frac{P^{n-2}(\sigma)}{Q^{n-2}(\sigma)} < \ldots < \frac{P'(\sigma)}{Q'(\sigma)} < \frac{P(\sigma)}{Q(\sigma)}$$

Putting $\sigma = 0$,

$$1 < \frac{a_{n-1}}{b_{n-1}} < \frac{a_{n-2}}{b_{n-2}} < \dots < \frac{a_1}{b_1} < \frac{a_0}{b_0}$$

b) RL Impedance or RC Admittance

Similarly, if $Z(s) = \dfrac{s^n + a_{n-1}s^{n-1} + \dots + a_0}{s^n + b_{n-1}s^{n-1} + \dots + b_0}$ is an RL impedance or RC admittance driving point function, it follows that

$$1 > \frac{a_{n-1}}{b_{n-1}} > \frac{a_{n-2}}{b_{n-2}} > \dots > \frac{a_1}{b_1} > \frac{a_0}{b_0}$$

c) LC Impedances and Admittances

Suppose $Z_{LC}(s) = \dfrac{s^{2n+1} + a_{2n-1}s^{2n-1} + a_{2n-3}s^{2n-3} + \dots + a_3s^3 + a_1s}{s^{2n} + b_{2n-2}s^{2n-2} + b_{2n-1}s^{2n-1} + \dots + b_2s^2 + b_0}$

is the LC driving point impedance function. As

$$Z_{RC}(p^2) = \frac{Z_{LC}(p)}{p}$$

$Z_{RC}(s)$, where $p^2 = s$, is an RC impedance function. It follows that

$$1 < \frac{a_{2n-1}}{b_{2n-2}} < \frac{a_{2n-3}}{b_{2n-4}} < \dots < \frac{a_3}{b_2} < \frac{a_1}{b_0}$$

Again, if $Z_{LC}(s)$ is the admittance function of an LC network,

$$1 < \frac{1}{a_{2n-1}} < \frac{b_{2n-2}}{a_{2n-3}} < \dots < \frac{b_2}{a_1} < \infty$$

The accompanying table summarizes the results of these sections a), b), c) above.

d) Positive Real Matrices

Theorem. A set of necessary conditions for

$$[\ell_{ij}(s)] = [\frac{P_{ij}(s)}{Q(s)}] = [\frac{s^n + a_{ij}s^{n-1} + b_{ij}s^{n-2} + \dots + k_{ij}}{s^n + a_0s^{n-1} + b_0s^{n-2} + \dots + k_0}] \quad \begin{array}{l} i=1,\dots,n \\ j=1,\dots,n \end{array}$$

to be a p.r. matrix is that the matrices

$$[\frac{k_{ij}}{k_0}] \;, \dots \; [\frac{b_{ij}}{b_0}] \;, \; [\frac{a_{ij}}{a_0}]$$

Type	P.R.F., Z(s)/K (where K is real constant greater than 0)	Coefficient Ordered Inequalities
RC i.p.r.f. or RL a.p.r.f.	(1) $\dfrac{s^n + a_{n-1}s^{n-1} + \dots + a_o}{s^n + b_{n-1}s^{n-1} + \dots + b_o}$	(1) $1 < \dfrac{a_{n-1}}{b_{n-1}} < \dfrac{a_{n-2}}{b_{n-2}} < \dots < \dfrac{a_1}{b_1} < \dfrac{a_o}{b_o}$
	(2) $\dfrac{a_{n-1}s^{n-1} + \dots + a_o}{s^n + b_{n-1}s^{n-1} + \dots + b_o}$	(2) $0 < \dfrac{a_{n-1}}{b_{n-1}} < \dfrac{a_{n-2}}{b_{n-2}} < \dots < \dfrac{a_1}{b_1} < \dfrac{a_o}{b_o}$
RC a.p.r.f. or RL i.p.r.f.	(3) $\dfrac{s^n + a_{n-1}s^{n-1} + \dots + a_o}{b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_o}$	(3) $\infty > \dfrac{a_{n-1}}{b_{n-1}} > \dfrac{a_{n-2}}{b_{n-2}} > \dots > \dfrac{a_1}{b_1} > \dfrac{a_o}{b_o}$
	(4) $\dfrac{s^n + a_{n-1}s^{n-1} + \dots + a_o}{s^n + b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_o}$	(4) $1 > \dfrac{a_{n-1}}{b_{n-1}} > \dfrac{a_{n-2}}{b_{n-2}} > \dots > \dfrac{a_1}{b_1} > \dfrac{a_o}{b_o}$
LC i.p.r.f.	(5) $\dfrac{s^{2n+1} + a_{2n-1}s^{2n-1} + \dots + a_3s^3 + a_1s}{s^{2n} + b_{2n-2}s^{2n-2} + \dots + b_2s^2 + b_o}$	(5) $1 < \dfrac{a_{2n-1}}{b_{2n-2}} < \dfrac{a_{2n-3}}{b_{2n-4}} < \dots < \dfrac{a_3}{b_2} < \dfrac{a_1}{b_o}$
	(6) $\dfrac{a_{2n-1}s^{2n-1} + a_{2n-3}s^{2n-3} + \dots + a_3s^3 + a_1s}{s^{2n} + b_{2n-2}s^2 + \dots + b_2s^2 + b_o}$	(6) $0 \quad \dfrac{a_{2n-1}}{b_{2n-2}} < \dfrac{a_{2n-3}}{b_{2n-4}} < \dots < \dfrac{a_3}{b_2} < \dfrac{a_1}{b_o}$
	(7) $\dfrac{s^{2n} + b_{2n-2}s^{2n-2} + \dots + b_2s^2 + b_o}{s^{2n+1} + a_{2n-1}s^{2n-1} + \dots + a_3s^3 + a_1s}$	(7) $0 < \dfrac{1}{a_{2n-1}} < \dfrac{b_{2n-2}}{a_{2n-3}} < \dots < \dfrac{b_2}{a_1} < \infty$
	(8) $\dfrac{s^{2n} + b_{2n-2}s^{2n-2} + \dots + b_2s^2 + b_o}{a_{2n-1}s^{2n-1} + \dots + a_3s^3 + a_1s}$	(8) $\dfrac{1}{a_{2n-1}} < \dfrac{b_{2n-2}}{a_{2n-3}} < \dots < \dfrac{b_2}{a_1} < \infty$

Type	P.R.F., Z(s)/K (where K is real constant greater than 0)	Coefficient Ordered Inequalities
LC a.p.r.f.	(9) $\dfrac{s^{2n+1} + a_{2n-1}s^{2n-1} + \dots + a_3s^2 + a_1s}{s^{2n} + b_{2n-2}s^{2n-2} + \dots + b_2s^2 + b_o}$	(9) $\alpha > \dfrac{a_{2n-1}}{1} > \dfrac{a_{2n-3}}{b_{2n-2}} \dots > \dfrac{a_3}{b_4} > \dfrac{a_1}{b_2}$
	(10) $\dfrac{a_{2n-1}s^{2n-1} + \dots + a_3s^2 + a_1s}{s^{2n} + b_{2n-2}s^{2n-2} + \dots + b_2s^2 + b_o}$	(10) $\dfrac{a_{2n-1}}{1} > \dfrac{a_{2n-3}}{b_{2n-2}} \dots > \dfrac{a_3}{b_4} > \dfrac{a_1}{b_2}$
	(11) $\dfrac{s^{2n} + b_{2n-2}s^{2n-2} + \dots + b_2s^2 + b_o}{s^{2n+1} + a_{2n-1}s^{2n-1} + \dots + a_3s^2 + a_1s}$	(11) $1 < \dfrac{a_{2n-1}}{b_{2n-2}} < \dots < \dfrac{a_3}{b_2} < \dfrac{a_1}{b_o}$
	(12) $\dfrac{s^{2n} + b_{2n-2}s^{2n-2} + \dots + b_2s^2 + b_o}{a_{2n-1}s^{2n-1} + \dots + a_3s^3 + a_1s}$	(12) $0 < \dfrac{a_{2n-1}}{b_{2n-2}} < \dots < \dfrac{a_3}{b_2} < \dfrac{a_1}{b_o}$

COEFFICIENT RELATIONS FOR P.R.F.'s REPRESENTING TWO-ELEMENT KIND NETWORKS.

be positive semidefinite, where elements $[\ell_{ij}(s)]$ do not have only common poles.

Proof:

If $[\ell_{ij}(s)]$ is a positive real matrix, then $[\ell_{ij}(0)] = [\frac{k_{ij}}{k_0}]$ is positive semidefinite. Using Theorems 1 and 2

$$[\ell^1_{ij}(s)], [\ell^2_{ij}(s)], \ldots , [\ell^{n-1}_{ij}(s)]$$

are also positive real matrices. It follows that $[\ell^1_{ij}(0)]$, $[\ell^{(2)}_{ij}(0)]$, ... $[\ell^{(n-1)}_{ij}(0)]$ are also positive semidefinite. Therefore, the proof of the theorem is complete.

REFERENCES

1. F. M. Reza, "Duex Theoremes sur les Dipoles E'lectriques," Comptes Rendus de l'Academie des Sciences, August 10, 1953, Vol. 237, pp. 429-430, Paris, France.

2. F. M. Reza, "A Generalization of Foster's and Cauer's Theorems," IRE Convention Record, 1955, Part 2, Circuit Theory, pp. 22-26.

3. A. Talbot, "Some Properties of Positive Functions," Trans. IEEE CT-12, December 1965, p. 607.

4. L. Weinberg and P. Stepian, "Realizability Conditions on n-Port Networks," IRE Trans. on Circuit Theory, September 1958.

5. N. K. Bose, Some New Results and Applications of Realizability Theory, Doctoral Dissertation, Syracuse University, 1967.

ON THE RELATION BETWEEN A POSITIVE REAL FUNCTION AND A THREE-PORT MATRIX

EZRA ZEHEB
Technion-Israel Institute of Technology, Haifa, Israel*

ABSTRACT

It is shown that any arbitrary rational positive real function can always be presented as an input impedance to a realizable reactive three-port, two ports of which are terminated in resistors.

Although this is a natural extension of Darlington's work, there are some differences between the two presentations, mostly in favor of the three-port.

Speculations are also made as to the possibilities of using these theorems to synthesize PR functions without transformers and with the minimum number of elements. An example is included.

I. INTRODUCTION

Darlington[1] has shown that any arbitrary rational positive real (PR) function can be presented as an input impedance to a realizable reactive two-port terminated in a single resistor. The objective of this paper is to show that such a function can also always be presented as an input impedance to a realizable reactive three-port, two ports of which are terminated in resistors.

Let the PR function be

$$Z(s) = \frac{m_1+n_1}{m_2+n_2} \tag{1}$$

where m_1 and m_2 are even polynomials and n_1 and n_2 are odd ones. Now augment the function by a Hurwitz polynomial m_0+n_0,

$$Z(s) = \frac{(m_1+n_1)\cdot(m_0+n_0)}{(m_2+n_2)\cdot(m_0+n_0)} = \frac{M_1+N_1}{M_2+N_2} \tag{2}$$

such that the new ensignant polynomial $M_1M_2 - N_1N_2$ will have all its zeros of even multiplicity. This is always possible since $M_1M_2 - N_1N_2 = (m_1m_2 - n_1n_2).(m_0^2 - n_0^2)$ and $m_1m_2 - n_1n_2$ has quadruple symmetry and if it has pure imaginary zeros, their multiplicity is even.

Darlington distinguishes between two cases according to whether $M_1M_2 - N_1N_2$ is even (case A) or odd (case B). If case A is relevant than the two matrices

---

* Presently at Stevens Institute of Technology, Hoboken, New Jersey on leave of absence from the Technion.

$$\begin{bmatrix} \dfrac{M_1}{N_2} & \dfrac{\sqrt{R(M_1M_2 - N_1N_2)}}{N_2} \\[2ex] \dfrac{\sqrt{R(M_1M_2 - N_1N_2)}}{N_2} & \dfrac{RM_2}{N_2} \end{bmatrix} \quad (a)$$

$$(3)$$

$$\begin{bmatrix} \dfrac{M_2}{N_1} & \dfrac{\sqrt{(M_1M_2 - N_1N_2)/R}}{N_1} \\[2ex] \dfrac{\sqrt{(M_1M_2 - N_1N_2)/R}}{N_1} & \dfrac{M_1}{RN_1} \end{bmatrix} \quad (b)$$

are realizable as reactive two-ports and each one of them, terminated in a resistor R, will yield Z(s) as input impedance. If case B is relevant, than replace the two matrices in equation (3) by the following ones:

$$\begin{bmatrix} \dfrac{N_1}{M_2} & \dfrac{\sqrt{R(N_1N_2-M_1M_2)}}{M_2} \\[2ex] \dfrac{\sqrt{R(N_1N_2-M_1M_2)}}{M_2} & \dfrac{RN_2}{M_2} \end{bmatrix} \quad (a)$$

$$(4)$$

$$\begin{bmatrix} \dfrac{N_2}{M_1} & \dfrac{\sqrt{(N_1N_2-M_1M_2)/R}}{M_1} \\[2ex] \dfrac{\sqrt{(N_1N_2-M_1M_2)/R}}{M_1} & \dfrac{N_1}{RM_1} \end{bmatrix} \quad (b)$$

In the next section we shall prove that two 3 x 3 matrices with similar properties are available. However, there are no two cases and each of these matrices may arbitrarily be chosen. Also, rationality of $\sqrt{M_1M_2-N_1N_2}$ is not required and there is never a need to augment the functions. The elements of these matrices are expressed in terms of any three of the polynomials m_1; m_2; n_1; n_2 (according to one's choice) in addition to two polynomials constructed from the ensignant $m_1m_2-n_1n_2$. Namely, the even and odd parts of a polynomial consisting of the left complex plane's zeros of $m_1m_2-n_1n_2$ and its pure imaginary zeros with half the multiplicity. (Obviously, the imaginary zeros of $m_1m_2-n_1n_2$ have even multiplicity).

In the third section we shall express some thoughts as to the connection between these mathematical theorems and the realization of PR functions without transformers and with a minimum number of elements. We shall also include an example.

II. THEOREMS AND PROOFS

Let

$$Z(s) = \frac{m_1+n_1}{m_2+n_2} \qquad (5)$$

where m_1 and m_2 are even polynomials and n_1 and n_2 are odd ones, be any positive real function. Define a Hurwitz polynomial H(s) by

$$H(s)\cdot H(-s) = m_1 m_2 - n_1 n_2 \qquad (6)$$

This is always possible since $m_1 m_2 - n_1 n_2$ has quadruple symmetry and if it has pure imaginary zeros, their multiplicity is even. Denote the even and odd parts of this Hurwitz polynomial by m and n respectively, i.e.

$$H(s) = m+n \qquad (7)$$

Let R_2 and R_3 be arbitrary positive real constants.
Then,

Theorem I:

$$(Z) = \begin{bmatrix} \dfrac{n^2 + m_1 m_2}{m_2 n_2} & \dfrac{\sqrt{R_2}\, m}{n_2} & \dfrac{\sqrt{R_3}\, n}{m_2} \\[2ex] \dfrac{\sqrt{R_2}\, m}{n_2} & \dfrac{R_2 m_2}{n_2} & 0 \\[2ex] \dfrac{\sqrt{R_3}\, n}{m_2} & 0 & \dfrac{R_3 n_2}{m_2} \end{bmatrix} \qquad (8)$$

is the open-circuit impedance matrix of a realizable reactive three-port which, when terminated at ports 2 and 3 with resistors R_2 and R_3 respectively, has an input impedance, looking into port 1, of Z(s). (Equation 5).

Proof:

We shall first prove that the input impedance is indeed Z(s) as given in Equation 5, and then proceed to prove the realizability of the three-port.

By definition,

$$(E) = (Z)\cdot(I) \qquad (9)$$

where (E) is a column matrix of the ports' voltages and (I) is a column matrix of the currents flowing into the ports. Also, when the three port is terminated as described in the theorem,

$$E_2 = -R_2 I_2; \qquad E_3 = -R_3 I_3 \tag{10}$$

From equations (9) and (10) we have:

$$\begin{bmatrix} E_1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} \\ Z_{12} & Z_{22} + R_2 & Z_{23} \\ Z_{13} & Z_{23} & Z_{33} + R_3 \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} \tag{11}$$

Therefore,

$$Z_{in} = \frac{E_1}{I_1} = \frac{\begin{vmatrix} Z_{11} & Z_{12} & Z_{13} \\ Z_{12} & Z_{22} + R_2 & Z_{23} \\ Z_{13} & Z_{23} & Z_{33} + R_3 \end{vmatrix}}{\begin{vmatrix} Z_{22} + R_2 & Z_{23} \\ Z_{23} & Z_{33} + R_3 \end{vmatrix}} = \frac{\begin{vmatrix} \frac{n^2 + m_1 m_2}{m_2 n_2} & \frac{\sqrt{R_2}\, m}{n_2} & \frac{\sqrt{R_3}\, n}{m_2} \\ \frac{\sqrt{R_2}\, m}{n_2} & \frac{R_2 m_2}{n_2} + R_2 & 0 \\ \frac{\sqrt{R_3}\, n}{m_2} & 0 & \frac{R_3 n_2}{m_2} + R_3 \end{vmatrix}}{\begin{vmatrix} \frac{R_2 m_2}{n_2} + R_2 & 0 \\ 0 & \frac{R_3 n_2}{m_2} + R_3 \end{vmatrix}} \tag{12}$$

Direct computation shows that:

$$Z_{in} = \frac{n^2 - m^2 + m_1 \cdot m_2 + m_1 n_2}{n_2 (m_2 + n_2)} \tag{13}$$

From equations (6) and (7) we have:

$$(m+n)\ (m-n) = m^2 - n^2 = m_1 m_2 - n_1 n_2 \tag{14}$$

so that

$$Z_{in} = \frac{m_1 + n_1}{m_2 + n_2}$$

which completes the first step of our proof.

Next, we want to prove that the three-port matrix (equation 8) is positive real and odd, and thus realizable with reactive elements only. Namely, we shall have to prove that:

1. The matrix is square and symmetric.
2. Its elements are odd rational functions of the complex frequency variable s, with real coefficients.
3. All poles of the matrix elements are pure imaginary and simple.
4. The matrix of the residues in every pole is real and positive semi-definite.

The first two points are obvious by observation.

As to the third point, the properties of a positive real (PR) function indicate that the zeros of m_2 and n_2, which are the finite poles of the matrix elements, are simple and pure imaginary. For a possible pole at infinity, however, we must look carefully at the degrees of the various polynomials involved. Let us denote the degree of m_2 by q:

$$\delta(m_2) = q \tag{15}$$

and construct a table of all possible degrees of the various polynomials. In doing this, we shall use the following properties of a PR function:

a) Both numerator m_1+n_1 and denominator m_2+n_2 are Hurwitz polynomials and differ in degrees by at most one.

b) The polynomial $m_1m_2 - n_1n_2$ has quadruple symmetry and thus,

$$\delta\left[H(s)\right] = 1/2 \;\delta\,(m_1m_2-n_1n_2). \tag{16}$$

The result is given in Table No. 1, from which it is readily verified that if any element of the matrix has a pole at infinity, it is simple.

	$\delta(m_2)$	$\delta(n_2)$	$\delta(m_1)$	$\delta(n_1)$	$\delta(m_1m_2-n_1n_2)$	$\delta[H(s)]$	$\delta(m)$	$\delta(n)$
1	n	n+1	n+2	n+1	$\leq 2n+2$	$\leq n+1$	$\leq n$	$\leq n+1$
2	n	n+1	n	n+1	2n+2	n+1	n	n+1
3	n	n+1	n	n-1	$\leq 2n$	$\leq n$	$\leq n$	$\leq n-1$
4	n	n-1	n	n+1	$\leq 2n$	$\leq n$	$\leq n$	$\leq n-1$
5	n	n-1	n	n-1	2n	n	n	n-1
6	n	n-1	n-2	n-1	$\leq 2n-2$	$\leq n-1$	$\leq n-2$	$\leq n-1$

Table No. 1

As to the residue matrix (point 4), we first claim realness. The residue of a function $\frac{P(s)}{Q(s)}$ in a finite pole $s = j\omega_0$ is given by $\left.\frac{P}{Q'}\right|_{s=j\omega_o}$ where Q' is the derivative of Q with respect to s. Hence, oddness of the functions in the matrix insures realness of the residues in the finite poles. The residue in a pole at infinity is given by $\left.\frac{P}{sQ}\right|_{s \to \infty}$. Hence, realness of the functions' coefficients insures realness of the residue in a pole at infinity.

Next, we claim "positive semi-definiteness". Denote the residue of Z_{ij} (i, j = 1, 2, 3) in any pole of Z_{ij} by K_{ij}, and denote the determinant of the residue matrix by Δ_K. By definition, the positive semi-definiteness of the matrix is assured if, and only if (recall that $K_{23}=0$), the following four conditions are satisfied:

$$K_{ii} \geqq 0 \quad (i = 1, 2, 3) \tag{17}$$

$$K_{11} K_{22} - K_{12}^2 \geqq 0. \tag{18}$$

$$K_{11} K_{33} - K_{13}^2 \geqq 0. \tag{19}$$

$$\Delta_K \geqq 0. \tag{20}$$

$Z_{22} = R_2 \frac{m_2}{n_2}$ and $Z_{33} = R_3 \frac{n_2}{m_2}$ are positive real odd (PRO) functions, since $Z(s) = \frac{m_1 + n_1}{m_2 + n_2}$ is PR. Therefore, K_{22} and K_{33} are necessarily positive for all poles. For a pole at infinity we have,

$$K_{11} = \left.\frac{n^2 + m_1 m_2}{s\, m_2\, n_2}\right|_{s \to \infty} \tag{21}$$

Since all the coefficients are positive we conclude that K_{11} is non-negative. It is finite because we have already proven that the multiplicity of a possible pole at infinity cannot exceed one.

$$K_{11} K_{22} - K_{12}^2 = \left(\left.\frac{n^2 + m_1 m_2}{s\, m_2\, n_2}\right|_{s \to \infty}\right)\left(\left.\frac{R_2 m_2}{s\, n_2}\right|_{s \to \infty}\right) - \left(\left.\frac{\sqrt{R_2}\, m}{s\, n_2}\right|_{s \to \infty}\right)^2$$

$$= R_2 \left.\frac{n^2 + m_1 m_2 - m^2}{s^2\, n_2^2}\right|_{s \to \infty} \tag{22}$$

Using equation (14), we get,

$$K_{11} K_{22} - K_{12}^2 = R_2 \left.\frac{n_1}{s^2\, n_2}\right|_{s \to \infty} \tag{23}$$

and since the coefficients of n_1 and n_2 are positive, we conclude that it is non-negative.

$$K_{11}K_{33} - K_{13}^2 = \left(\frac{n^2 + m_1 m_2}{s\, m_2\, n_2}\Bigg|_{s\to\infty}\right)\left(R_3 \frac{n_2}{m_2}\Bigg|_{s\to\infty}\right) - \left(\sqrt{R_3}\,\frac{n}{m_2}\Bigg|_{s\to\infty}\right)^2$$

$$= R_3 \frac{m_1}{s^2\, m_2}\Bigg|_{s\to\infty} \geq 0 \tag{24}$$

and,

$$\Delta_K = K_{33}\left(K_{11}K_{22} - K_{12}^2\right) - K_{13}^2 K_{22} = R_2 R_3 \frac{n_1 n_2 - n^2}{s^3\, m_2\, n_2}\Bigg|_{s\to\infty} \tag{25}$$

Using Table No. 1, it is easily verified that:

$$\Delta K = 0 \tag{26}$$

For a finite pole we have,

$$K_{11} = \frac{n^2 + m_1 m_2}{(m_2 n_2)'}\Bigg|_{s = j\omega_i \text{ or } s = j\omega_j} \tag{27}$$

where $s = j\omega_i\ (i = 1, 2, \ldots)$ are poles of Z_{11} contributed by the zeros of m_2 and $s = j\omega_j\ (j = 1, 2, \ldots)$ are those contributed by the zeros of n_2. We shall use the following property of PRO functions:

Let F(s) be a PRO function. Then,

$$\frac{dF}{ds}\Bigg|_{s = j\omega} > 0 \text{ for all } \omega. \tag{28}$$

Since $\frac{n_2}{m_2}$ is PRO, we have,

$$\frac{m_2 n_2' - n_2 m_2'}{m_2^2}\Bigg|_{s = j\omega} > 0 \tag{29}$$

Since m_2 is even, $m_2^2\Big|_{s = j\omega}$ is positive and hence,

$$m_2 n_2' - n_2 m_2'\Big|_{s = j\omega} > 0 \quad \forall\, \omega \tag{30}$$

At $\omega = \omega_i$, where m_2 vanishes, we get therefore:

$$n_2 m_2'\Big|_{s = j\omega_i} < 0 \tag{31}$$

while at $\omega = \omega_j$, where n_2 vanishes, we get:

$$m_2 n_2'\Big|_{s = j\omega_j} > 0. \tag{32}$$

Hence,

$$K_{11} = \left.\frac{n^2 + m_1 m_2}{(m_2\, n_2)'}\right|_{s = j\omega_i} = \left.\frac{n^2}{m_2\, n_2' + n_2\, m_2'}\right|_{s = j\omega_i}$$

$$= \left.\frac{n^2}{n_2\, m_2'}\right|_{s = j\omega_i} \qquad (33)$$

By equation (31), and since n is an odd function of s, K_{11} in the poles $s = j\omega_i$ ($i = 1, 2, \ldots$) is a ratio of real non-positive numbers and hence is non-negative.

Using equation (14), Z_{11} can also be expressed by

$$Z_{11} = \frac{m^2 + n_1 n_2}{m_2\, n_2} \qquad (34)$$

so that

$$K_{11} = \left.\frac{m^2 + n_1 n_2}{(m_2 n_2)'}\right|_{s = j\omega_j} = \left.\frac{m^2}{m_2\, n_2'}\right|_{s = j\omega_j} \qquad (35)$$

By equation (32), and since m is an even function of s, K_{11} in the poles $s = j\omega_j$ ($j = 1, 2, \ldots$) is a ratio of real non-negative numbers and is non-negative itself.

As to $K_{11}\, K_{22} - K_{12}^2$, we need only to consider poles contributed by zeros of n_2. For these,

$$K_{11}K_{22} - K_{12}^2 = \left(\left.\frac{m^2}{m_2\, n_2'}\right|_{s = j\omega_j}\right)\left(\left.\frac{R_2\, m_2}{n_2'}\right|_{s = j\omega_j}\right) - \left(\left.\frac{R_2\, m}{n_2'}\right|_{s = j\omega_j}\right)^2 = 0. \qquad (36)$$

For $K_{11}\, K_{33} - K_{13}^2$, we need only to consider poles contributed by zeros of m_2.

$$K_{11}K_{33} - K_{13}^2 = \left[\left(\frac{n^2}{n_2\, m_2'}\right)\cdot\left(\frac{R_3\, n_2}{m_2'}\right) - \frac{R_3 n^2}{(m_2')^2}\right]_{s = j\omega_i} = 0 \qquad (37)$$

As for $\Delta_K = K_{33}\,(K_{11}\,K_{22} - K_{12}^2) - (K_{13})^2\, K_{22}$, it has just been proven that the first expression vanishes for all finite poles. Noticing that Z_{13} and Z_{22} have no common poles, we verify that the second expression vanishes too, and thus $\Delta_K = 0$ for all finite poles as well as for the pole at infinity.

This completes the proof of Theorem I.

Theorem II.

$$Y = \begin{bmatrix} \frac{n^2 + m_1 m_2}{m_1 n_1} & \frac{m}{\sqrt{R_2}\, n_1} & \frac{n}{\sqrt{R_3}\, m_1} \\ \frac{m}{\sqrt{R_2}\, n_1} & \frac{m_1}{R_2 n_1} & 0 \\ \frac{n}{\sqrt{R_3}\, m_1} & 0 & \frac{n_1}{R_3 m_1} \end{bmatrix} \qquad (38)$$

is the short-circuit admittance matrix of a three-port such as in Theorem I.

The proof of this theorem follows the same lines as in the proof of the first theorem, and will, for the sake of brevity, be omitted.

III. POSSIBLE APPLICATIONS

It is worthwhile to speculate on the possibilities of using the theorems which were proven here, to synthesyze PR functions without transformers and with the minimum number of elements.

Seshu[2] has conjectured that minimal realization without transformers would require, in addition to reactive elements, no more than two resistors. (Obviously, no less than two resistors are necessary if $0 < Z(0) \neq Z(\infty) < \infty$. Except for biquadratic functions, this conjecture has not yet been proven or disproven. It has not even been proven that a PR function can at all be realized without transformers and with no more than two resistors. Nevertheless, it has been proven by the author together with A. Lempel[3, 4] that given a finite set of n arbitrary real frequencies in which a function has to receive n arbitrary complex values $R_i + jX_i$ (i = 1, 2, ..., n), conditioned only by $R_i \geq 0$ for all i, it is always possible to find a network consisting of inductors, capacitors and no more than two resistors, the input impedance of which complies with the requirements. This suggests that Seshu's conjecture is more likely to be proven than disproven. Therefore, there is good reason to search for a transformerless realization of at least one of our three-ports, which will lead to the realization of a PR function without transformers and with a minumum number of elements. (Recall that the function is not augmented). It is not difficult to find examples of PR functions which, when realized by Darlington's procedure, necessarily require transformers, whereas by the three-port procedure can be realized without transformers. For example, choose:

$$Z_{in} = \frac{s^3 + 2s^2 + 3/2s + 2}{s^3 + s^2 + 2s + 1} \qquad (39)$$

Since $0 < z(\infty) \neq z(0) < \infty$, this impedance cannot be realized without transformers by the Darlington procedure, no matter which augmentation will be utilized.

Now turn to the three-port procedure and construct first the polynomial H(s).

$$m_1 = 2(s^2 + 1) \tag{40}$$

$$n_1 = s(s^2 + 3/2) \tag{41}$$

$$m_2 = s^2 + 1 \tag{42}$$

$$n_2 = s(s^2 + 2) \tag{43}$$

$$H(s) \cdot H(-s) = m_1 m_2 - n_1 n_2 = -s^6 - 3/2\, s^4 + s^2 + 2$$

$$= (s^3 + \frac{\sqrt{2}}{2} s^2 + s + \sqrt{2}) \cdot (-s^3 + \frac{\sqrt{2}}{2} s^2 - s + \sqrt{2}) \tag{44}$$

Hence,

$$H(s) = s^3 + \frac{\sqrt{2}}{2} \cdot s^2 + s + \sqrt{2} \tag{45}$$

and,

$$m = \frac{\sqrt{2}}{2} (s^2 + 2); \qquad n = s(s^2 + 1) \tag{46}$$

Using theorem I, the three-port matrix is

$$(Z) = \begin{pmatrix} \frac{s^2+1}{s} & \sqrt{R_2}\,\frac{\sqrt{2}}{2}\,\frac{1}{s} & \sqrt{R_3}\; s \\ \sqrt{R_2}\,\frac{\sqrt{2}}{2}\,\frac{1}{s} & R_2\,\frac{s^2+1}{s\,(s^2+2)} & 0 \\ \sqrt{R_3}\; s & 0 & R_3\,\frac{s(s^2+2)}{s^2+1} \end{pmatrix} \tag{47}$$

With some experience in realizing three-ports, it is readily seen that choosing

$$R_2 = 2\,\Omega; \qquad R_3 = 1\,\Omega \tag{48}$$

yields the network shown in Figure 1.

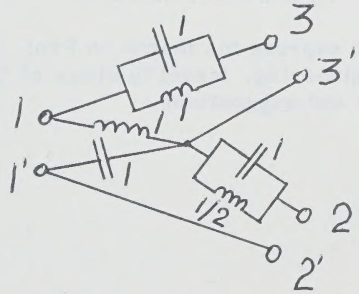


Fig. 1. The reactive three-port.

Direct analysis shows that the input impedance when terminating ports 2 and 3 in resistors 2Ω and 1Ω respectively, is

$$Z_{in} = \frac{s^3 + 2s^2 + 3/2\ s + 2}{s^3 + s^2 + 2s + 1} \tag{49}$$

as should be.

IV. CONCLUSIONS

It has been shown that any arbitrary rational positive real function can always be presented as an input impedance to a realizable reactive three-port, two ports of which are terminated in resistors.

Although this is a natural extension of Darlington's work, there are some differences between the two presentations, mostly in favor of the three-port.

In Darlington's procedure, the PR function has, in general, to be augmented by a certain Hurwitz polynomial before the two-ports matrix can be determined. The reason is, that a necessary condition for the matrix realizability is that the ensignant polynomial should have all its zeros of even multiplicity, which is not necessarily true for an arbitrary PR function. In the present procedure, the fulfillment of this condition is not required and hence no augmentation is ever needed.

Also, there is no sub-division into two "cases". Each of the two different three-port matrices can arbitrarily be chosen for any PR function.

Another difference, which puts the three-port presentation at a slight disadvantage, is that some of its matrix' elements are expressed in terms of polynomials constructed from the ensignant polynomial, whereas the elements of the two-port matrix do not require this additional construction.

It has been argued that the three-port presentation may contribute to the solution of the problem of economical realization of rational PR functions without transformers.

ACKNOWLEDGEMENT

The author wishes to express his thanks to Prof. I. Navot of the Faculty of Electrical Engineering, Israel Institute of Technology, for many helpful discussions and suggestions.

REFERENCES

1. S. Darlington, "Synthesis of Reactance 4-Poles which Produce Prescribed Insertion-Loss Characteristics," J. Math. and Phys., Vol. 18, (1939).
2. S. Seshu, "Minimal Realizations of the Biquadratic Minimum Function," Trans. IRE on C. T., Vol. CT-6, Dec. 1959.
3. E. Zeheb and A. Lempel, "Interpolation in the Network Sense," IEEE Trans. on C. T., Vol. CT-13, March, 1966.
4. A. Lempel and E. Zeheb, "A Note on Interpolation," To be published in IEEE Trans. on C. T.

ON THE DETECTION CAPABILITY OF NONBINARY CODES USING PERMUTATION MAPPINGS

D. T. TANG and V. Y. LUM
IBM Watson Research Center, Yorktown Heights, New York

ABSTRACT: In this paper error detection schemes for nonbinary systems are studied. To achieve maximum detection with minimum redundancy, permutation functions are used in check equations. Conditions are derived under which the detection of all single errors and adjacent switch-overs can be accomplished with one check digit. With an additional digit, all single deletion and insertion errors can also be detected.

INTRODUCTION

Although binary representation of information is typical of today's digital communication and computing systems, there remain situations where nonbinary representations are still needed in certain parts of the overall system. Decimal or alpha-numeric terminals, for instance, are invariably used in credit card, banking or similar systems. Information such as an ID number, when keyed in, is always subject to possible error. A certain degree of protection by means of coding therefore becomes desirable. Several features are important whenever a checking scheme is used: (1) It must be simple, and there should be as few check digits as possible. (2) It should be applicable to code words of variable length. (3) It should be sensitive to certain type of errors which are known to occur much more often than others, namely single errors, switch-over errors of adjacent symbols, and deletion or insertion of a symbol. This paper presents the basic results on the detection capability of certain very simple schemes.

POLYNOMIAL CODES

In the direct application of linear codes of q-nary systems, the q is limited to prime powers. In practice, the q may not satisfy this condition as is the case of a decimal system. One way to circumvent this difficulty is to represent each symbol in terms of its respective residues modulo the prime (or prime power) factor of q. A code based on each factor is next constructed and the superimposed code is obtained. All the calculations regarding the q-nary code are actually carried out modulo each individual factor, according to its corresponding parity-check matrix. It can be easily shown that the capability of such a code is the least and the length is the shortest among its factor codes.

Clearly the above method is most suitable for codes of some fixed length. Furthermore, except for the fixed length codes where "end of code word" is always transmitted, these codes can not detect any deletion or insertion errors.

CODING METHOD WITH PERMUTATION MAPPINGS

Consider now the problem of detecting single switch-over errors in addition to single errors. We shall derive conditions under which the desired detection capabilities can be accomplished using only one check digit.

Let us consider a check equation in the general form

$$f_1(a_1) + f_2(a_2) + \ldots + f_k(a_k) + f_{k+1}(c) \equiv 0, \quad \text{mod } q \qquad (1)$$

where $f_i(x)$ are functions mapping the set of code symbols to itself, a_i is the information symbol in the i^{th} digit, c the check symbol, and q the radix of the system (number of code symbols). Hereafter, a congruence sign, when used alone, will be understood to imply modulo q operation. The main results in this section will be presented in the form of several theorems.

Theorem 1 - For a single check equation to detect all single errors, each $f_i(a)$, $i=1, 2, \ldots, k+1$ must be a permutation function on the set of code symbols, $A = \{0, 1, \ldots, q-1\}$.

Proof - To detect all single errors, $f_i(a_1)$ must not be the same as $f_i(a_2)$ for any $a_1 \neq a_2$, where $a_1, a_2 \in A$. Hence $f_i(a)$ must be a permutation function.

Eq. (1) can now be written as

$$p_1(a_1) + p_2(a_2) + \ldots p_k(a_k) + p_{k+1}(c) \equiv 0 \qquad (2)$$

where p_i are permutation functions.

So far, no detection of switch-over errors has been considered. To detect switch-over errors, $p_i(a)$ must not be the same as $p_{i+1}(a)$. This is a necessary condition but not a sufficient one. Before we go on to derive the necessary and sufficient condition of single switch-over error detection, we need the following definition.

Definition - A permutation $p(a) = \overline{p}(a)$ is a "perfect-difference permutation" (PDP) if $\overline{p}(a_1) - a_1 \neq \overline{p}(a_2) - a_2$ for any $a_1 \neq a_2$, where $a_1, a_2 \in A$.

A basic property of PDP's will be stated in the following theorem which needs no proof.

Theorem 2 - If $\bar{p}(a)$ is a PDP, so are $\bar{p}^{-1}(a)$ and $\bar{p}(a) + m$, where $\bar{p}^{-1}(a)$ is the inverse mapping of $\bar{p}(a)$, and $m \in A$ is a constant.

We shall next show that permutation and PDP's have corresponding graphs with very interesting properties. Let each of the symbols in A represent a node. Then a directed edge can be drawn from a_i to a_j if $p(a_i) = a_j$. Since p is a permutation, each node will be incident to two edges, one directed toward and one directed away from the node. An immediate result is that this graph, to be called G_p, the permutation graph of p(a), is a disjoint union of cycles. Let us denote these cycles by $C_1, C_2, \ldots, C_s$, and define the distance of an edge b_k from a_i to a_j to be $d_{ij} \equiv a_j - a_i$. Alternatively, the distance of an edge b_k can be written as $d(b_k)$. Clearly, in a PDP graph, distinct edges must have distinct distances. It follows immediately that when the distances of all the edges in a cycle are summed, each node number a_i on the cycle will appear twice in the summation with opposite signs, thus cancelling each other. That is,

$$\sum_{b_j \in C_i} d(b_j) \equiv 0 \qquad (3)$$

for each C_i, and thus

$$\sum_{b_j \in G_p} d(b_j) = \sum_{C_i} \sum_{b_j \in C_i} d(b_j) \equiv 0 \qquad (4)$$

for every permutation graph G_p.

Let us now consider an illustrative example. Suppose we have q = 5. The permutations in Figure 1 are PDP's with their corresponding graphs shown in Figure 2.

The next theorem states the capability of a code using PDP mappings.

Theorem 3 - A code with check equation as in Eq. (2) will detect single switch-over errors if and only if the mappings between permutation on adjacent positions, i.e. $p_{i+1}(p_i^{-1}(a))$, $i=1, 2, \ldots, k$, are all PDP's.

Proof - Let two adjacent symbols a_i, a_{i+1} be interchanged. To detect this error, the check equation must not be zero after the interchange, that is,

$$p_i(a_i) + p_{i+1}(a_{i+1}) \not\equiv p_i(a_{i+1}) + p_{i+1}(a_i) \qquad (5)$$

Substituting $b = p_i(a)$, we obtain

$$b_i + p_{i+1}(p_i^{-1}(b_{i+1})) \not\equiv b_{i+1} + p_{i+1}(p_i^{-1}(b_i)) \qquad (6)$$

By definition, $p_{i+1}(p_i^{-1}(b))$ is a PDP.

Conversely, if $p_{i+1}(p_i^{-1}(a))$ for $i=1, 2, \ldots, k$ are PDP's, then Eqs. (6) and (5) follow, and therefore all single switch-over errors will be detected.

At a glance, one may think that it would be generally difficult to find enough permutations which satisfy conditions of Theorems 1 and 3. However, simple observation reveals that the existence of a single PDP, $\bar{p}(a)$, is sufficient. To show this, let $p_i(a_i) = a_i$, for all odd i, and let $p_i(a_i) = \bar{p}(a_i)$ for all even i. The check equation becomes

$$\left.\begin{array}{ll} a_1 + \bar{p}(a_2) + a_3 + \ldots + a_k + \bar{p}(c_1) \equiv 0 & \text{for k odd} \\ \text{and} & \\ a_1 + \bar{p}(a_2) + a_3 + \ldots + \bar{p}(a_k) + c_1 \equiv 0 & \text{for k even} \end{array}\right\} \qquad (7)$$

The existence of PDP's is considered in the following theorems.

Theorem 4 - No PDP exists for q even.

Proof - We have noted earlier that in a PDP graph, distinct edges must have distinct distances. Since there are exactly q edges, we must have

$$\sum_{b_j \in G_{\bar{p}}} d(b_j) = 0 + 1 + \ldots + (q-1) = \frac{q(q-1)}{2} \equiv 0 \qquad (8)$$

from Eq. (4), which requires $\frac{q(q-1)}{2}$ be divisible by q. It is easy to see that, if q is even, and thus q-1 is odd, this condition cannot be satisfied.

Theorem 5 - PDP exists for q odd.

Proof - Although Eq. (8) is satisfied for q odd, it is only a necessary but not a sufficient condition. To show that PDP exists for q odd we shall rely on a constructive proof.

Consider the permutation graph of $p(i) = q-i$ for $i \in A$. The edge directed from i to q-i has a distance $q-i-i \equiv -2i$.

For two edges to have the same distance,

$$-2i + 2j \equiv 2(j-i) \equiv 0 \qquad (9)$$

Since 2 is relatively prime with the odd q, Eq. (9) is satisfied if and only if i=j. Thus all distances are distinct and the permutation so defined is a PDP.

Let us now define a linear permutation p to be one which satisfies $p(a) \equiv ma$, where $a, m \in A$. Linear permutations are nice because they imply simple arithmetic operations.

Theorem 6 - If $p(a) \equiv ma$ is a PDP, then $p'(a) \equiv (m-1)a$ is a permutation function. Conversely, if $p(a) \equiv ma$ and $p'(a) \equiv (m-1)a$ are permutations, then p is a PDP.

Proof - If p(a) is a PDP, then

$$p(a_i) - a_i \neq p(a_j) - a_j, \quad \text{for } i \neq j \tag{10}$$

For $p(a) \equiv ma$, Eq. (10) becomes

$$ma_i - a_i \neq ma_j - a_j \tag{11}$$

or

$$(m-1)a_i \neq (m-1)a_j \tag{12}$$

It follows that $p'(a) \equiv (m-1)a$ is a permutation.

Conversely, if $p'(a) \equiv (m-1)a$ is a permutation, then Eq. (12) holds for $a_i \neq a_j$. Eqs. (11) and (10) then follow and therefore $p(a) \equiv ma$ is a PDP.

Corollary 1 - The function $f(a) \equiv ma$ is a PDP if and only if q, m, m-1 are pair-wise relatively prime.

Corollary 2 - Linear PDP exists for $q \geq 3$ and odd.

Proof - Let m = 2, then m-1 = 1. By Corollary 1 $p(a) \equiv 2a$ is a PDP. Similarly, $p(a) \equiv (q-1)a$ is another PDP which is described in the constructive proof of Theorem 5. Thus for $q > 3$ and odd, there are at least two distinct PDP's.

For q an odd prime, all linear permutations except the identity, are PDP's. The converse of this statement is not true. This is illustrated in Figure 3.

For q even, Theorem 4 states that no PDP exists. However, one may still be interested in finding a scheme that will detect as many single switch-over errors as possible. The following theorem clarifies the situation in this respect.

Theorem 7 - For q even, there exists a check scheme in the form of Eq. (7) such that all single switch-over errors are detected except when the switch-over is between a specific pair a_1 and a_2.

Proof - It suffices to show that there exists a permutation p(a) such that the condition $p(a_i) - a_i \neq p(a_j) - a_j$ is satisfied for all $a_i, a_j \epsilon A$ except for a specific pair a_1 and a_2. It can easily be seen that, for q = 2, the only confusion pair is (0, 1). For q > 2 and even, we consider the following two cases.

If q = 4m, let the permutation p be given by:

$$\left.\begin{aligned} p(a) &= a, \quad \text{for } a = 0, \frac{q}{4} \\ p(a) &= q-a, \text{ for } 0 < a < \frac{q}{4} \text{ and } \frac{3q}{4} < a < q \\ p(a) &= q+1-a, \text{ for } \frac{q}{4} < a \leq \frac{3q}{4} \end{aligned}\right\} \tag{13}$$

If q = 4m+2, let p be given by:

$$\left.\begin{aligned} p(a) &= a, \quad \text{for } a=0, \frac{q+2}{4} \\ p(a) &= q-a, \text{ for } 0 < a < \frac{q+2}{4} \text{ and } \frac{3q-2}{4} < a < q \\ p(a) &= q+1-a, \text{ for } \frac{q+2}{4} < a \leq \frac{3q-2}{4} \end{aligned}\right\} \tag{14}$$

The graphs of these permutations are illustrated for the cases q=8 and q=10 in Figure 4. It is clear from the corresponding graphs that the edges in each graph all have distinct distances except for the pair 0 and $\frac{q}{4}$ if q=4m, or 0 and $\frac{q+2}{4}$ if q = 4m+2. These pairs constitute the only undetectable single switch-over errors.

It should be pointed out that since there often exist more than one permutation which satisfy Theorem 7, one can choose a permutation in which the undetectable switch-over error occurs to the pair of symbols least likely to be confused. In the case of q=10, it has been found that the only undetectable switch-over may be made to occur to any prescribed pair of symbols. Note that the permutations here are always nonlinear.

DETECTION OF DELETION AND INSERTION ERRORS

Suppose that we wish to detect single-digit deletion and insertion errors in addition to the single and single switch-over errors with two check digits c_1 and c_2, where c_1 is defined by Eq. (7) and c_2 is to be added at the end of the code word after c_1. If c_2 is deleted, then c_1

will be taken as c_2 and a_k as c_1. The first check will be satisfied if and only if $c_1 = 0$ for k even or $\bar{p}(c_1) = 0$ for k odd. On the other hand, if a digit is inserted after c_2, then c_2 will be taken as c_1. Here again the first check will be satisfied if $c_2 = 0$ for k odd, or $\bar{p}(c_2) = 0$ for k even.

In light of the above observations, let us choose the PDP in Eq. (7) to be one such that $\bar{p}(0) = 0$. Let c_2 be 1 if k is odd and 2 if k is even. Since single errors, single switch-overs, single-digit deletions not involving c_2, or single-digit insertions before c_2 will clearly be detectable, the only cases left consist of a deletion of c_2 or an insertion after c_2. The latter case is detectable since c_2, $\bar{p}(c_2) \neq 0$ implies that the first check must fail. In the former case where c_2 is deleted, the first check will be satisfied only if $c_1 = 0$. Therefore, either this check will fail, thus indicating an error, or we must have $c_1 = \bar{p}(c_1) = 0$. But c_1 is taken as c_2 and both c_2 and $\bar{p}(c_2)$ cannot be zero. The error is therefore also detectable.

DISCUSSION AND CONCLUSION

There are feasible variations and extensions of the coding schemes described in this paper. For instance, one may choose to use an expanded symbol set for check digits and thus essentially work with a prime power q. The inclusion of a "hyphen" (which is usually available) in addition to the ten decimal figures, for example, enables one to work in the field of 11 elements for which better linear codes exist. The use of more check equations with permutation functions to attain higher detection capability is another possibility.

The detection of deletion and insertion errors by an additional digit as described in the last section seems very interesting. It should be pointed out that such a check digit can always be used in conjunction with any other nonbinary checking scheme in which the last check equation takes the general form of Eq. (1).

Credit card systems with decimal key-punch terminals using one check symbol to detect single and most of the single switch-over errors have been in existence. It is believed that the improved scheme as suggested in this paper can further enhance the reliability of such systems and will find applications in other similar systems in the future.

ACKNOWLEDGMENT

The authors would like to thank Miss Fran Youngstein of IBM for introducing the problem to them and supplying useful information regarding the existing credit card systems.

BIBLIOGRAPHY

1. Peterson, W. W., Error Correcting Codes, Cambridge, Mass., MIT Press, 1961.
2. van der Waerden, B. L., Modern Algebra, Vol. 1, New York, Unger, 1949.

a	0	1	2	3	4
$\bar{p}(a)$	0	2	4	1	3

(a)

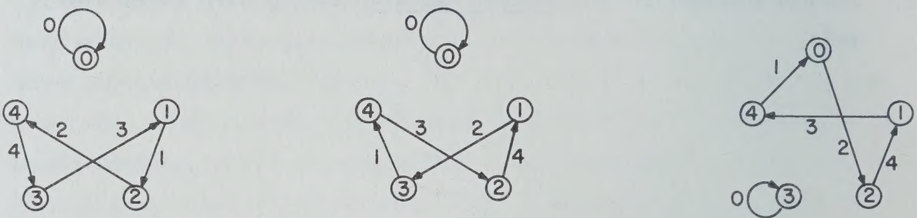
a	0	1	2	3	4
$\bar{p}^{-1}(a)$	0	3	1	4	2

(b)

a	0	1	2	3	4
$\bar{p}(a)+2$	2	4	1	3	0

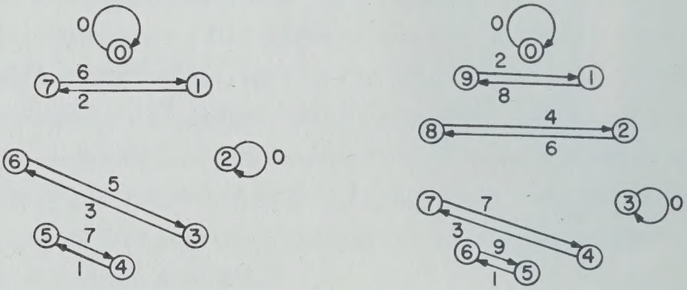
(c)

FIGURE 1


0
0
4
1
3
2
2
3
4
1
0
0
4
1
3
2
3
2
1
4
0
1
4
1
3
2
3
4
0
2

(a) $\bar{p}(a)$ (b) $\bar{p}^{-1}(a)$ (c) $\bar{p}(a)+2$

FIGURE 2


0
0
7
1
6
2
6
3
5
3
2
0
5
4
7
1
0
0
9
1
2
8
8
2
4
6
7
4
7
3
6
5
9
1
3
0

(a) $q = 8$ (b) $q = 10$

FIGURE 3

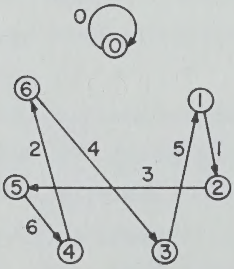

0
0
6
1
2
4
5
1
5
2
3
6
4
3

FIGURE 4. A NONLINEAR PDP FOR $q = 7$

DISTANCE MEASURES AND RELATED CRITERIA

HISASHI KOBAYASHI AND JOHN B. THOMAS
Princeton University, Princeton, New Jersey

ABSTRACT

Distance measures and related criteria in estimation and detection are discussed, and the Bhattacharyya distance and J-divergence are studied in detail. One of the results is that the ratio of the distance measures of two test statistics is equivalent to asymptotic relative efficiency. Two-input systems are discussed as examples.

1. SOME DISTANCE MEASURES AND THEIR COMMON PROPERTIES

The notion of a distance measure between two probability measures is widely used in statistics. Grettenberg [1], Kailath [2], and Kadota and Shepp [3] discuss the application of some of these measures to communication problems. In this treatment distance measures and related criteria in statistical communication theory are considered.

Let $p_1(x)$ and $p_2(x)$ be density functions of probability measures $P_1(x)$ and $P_2(x)$ defined over X, a space of observations x, under the hypotheses H_1 and H_2 respectively. Let L(x) be the Radon-Nikodym derivative of P_2 with respect to P_1, i.e.,

$$L(x) = \frac{p_2(x)}{p_1(x)} \tag{1.1}$$

Then many of the distance measures currently used can be written in the form [4]

$$f\{E_1[\phi(L)]\} \tag{1.2}$$

where $\phi(\cdot)$ is a continuous convex function on $(0,\infty)$, $f(\cdot)$ is an increasing real-valued function of a real variable, and $E_1[\cdot]$ is the expectation under the probability measure P_1. Typical examples are:

(a) Kullback-Leibler numbers [5]

$$I(1,2) = \int p_1(x) \ln \frac{p_1(x)}{p_2(x)} dx = E_1[- \ln L] \tag{1.3}$$

$$I(2,1) = \int p_2(x) \ln \frac{p_2(x)}{p_1(x)} dx = E_1[L \ln L] \tag{1.4}$$

(b) J-divergence [5]

$$J = I(1,2) + I(2,1) = E_1[(L-1) \ln L] \tag{1.5}$$

(c) Matsushita's measure of distance [6]

$$d = [\int(\sqrt{p_2(x)} - \sqrt{p_1(x)})^2 dx]^{1/2} = [E_1(\sqrt{L}-1)^2]^{1/2} \tag{1.6}$$

(d) Kolmogorov's measure of variational distance

$$K = \frac{1}{2} \int |p_2(x) - p_1(x)| \, dx = \frac{1}{2} E_1|L - 1| \tag{1.7}$$

(e) Bhattacharyya distance (B-distance) [2]

$$B = - \ln \rho \tag{1.8}$$

where ρ is defined by

$$\rho = \int\sqrt{p_1(x)\ p_2(x)}\ dx = E_1[\sqrt{L}\ ] \qquad (1.9)$$

and is called the Bhattacharyya coefficient [2], or Hellinger integral [3], or affinity [6].

An important theorem can now be derived which relates the Bayes risk to quantities of the form of Eq. (1.2) and which has direct application to signal selection problems: Let $\pi = (\pi_1, \pi_2)$ be the prior probability set where π_i is the prior probability of the hypothesis H_i, $i = 1,2$. Let C_{ij} be the cost assigned if H_j is chosen when H_i is true. Let X_i be a subspace of X such that $X_i = \{x | H_i \text{ is chosen when } x \text{ observed}\}$, $i = 1,2$. If the criterion is minimization of average risk, then by the well-known Neyman-Pearson lemma, X_i should be chosen as follows:

$$X_1 = \{x | L(x) < \frac{\pi_1(C_{12} - C_{11})}{\pi_2(C_{21} - C_{22})}\} \text{ and } X_2 = X \sim X_1$$

Let α and β be two signal sets (or communication systems, in general) which are to be compared under the criterion of the Bayes risk. The Bayes risk when the signal α is adopted, is given by

$$R(\alpha,\pi) = \pi_1 C_{12} + \pi_2 C_{22} + \int_{X_{\alpha 1}} [\pi_2 C_2' L_\alpha(x) + \pi_1 C_1'] p_{\alpha 1}(x)\, dx \qquad (1.10)$$

where $C_1' = C_{11} - C_{12}$, $C_2' = C_{21} - C_{22}$ and where $X_{\alpha 1}$, $L_\alpha(x)$, etc. are self-explanatory. On defining $R(\beta,\pi)$ in exactly the same way, the following theorem can be stated.

Theorem 1

$R(\alpha,\pi) \leq R(\beta,\pi)$ for all sets of prior probability $\pi = (\pi_1, \pi_2)$ if, and only if

$$E_1[\phi(L_\alpha)] \geq E_1[\phi(L_\beta)] \qquad (1.11)$$

for all continuous convex functions $\phi(\cdot)$.

Proof

The proof will be found elsewhere [8] and can be obtained by generalizing the argument given by Karlin and Bradt [7].//

On letting $P_e(\alpha,\pi)$ and $P_e(\beta,\pi)$ be the error probability under the signal sets α and β, respectively, the following statement can be derived from Theorem 1:

Corollary 1

$P_e(\alpha,\pi) \leq P_e(\beta,\pi)$ for all sets of prior probability $\pi = (\pi_1, \pi_2)$ if and only if

$$E_1[\phi(L_\alpha)] \geq E_2[\phi(L_\beta)]$$

for all continuous convex functions $\phi(\cdot)$.

Proof

See [7,8].

By contrapositive relations, Theorem 1 and Corollary 1 are equivalent to the following:

Corollary 2

$R(\alpha,\pi) > R(\beta,\pi)$ or, equivalently, $P_e(\alpha,\pi) > P_e(\beta,\pi)$ for some set of prior probability $\pi = (\pi_1, \pi_2)$ if and only if

$$E_1[\phi(L_\alpha)] < E_1[\phi(L_\beta)]$$

for some continuous convex function $\phi(\cdot)$.

Grettenberg [1] and Kailath [2] applied Corollary 2 to the cases of the J-divergence and the B-distance, respectively and discussed the signal selection problem.

2. THE BHATTACHARYYA COEFFICIENT AND THE BAYES RISK

It will be shown in this section that an upper and lower bound of the Bayes risk is given in terms of the Bhattacharyya coefficient ρ, which is a generalization of the inequality relation for the probability of error given by Kailath [2]. The Bayes Risk under the probability set π is

$$R(\pi) = \pi_1 C_{12} + \pi_2 C_{22} + \int_{X_1} [\pi_2 C_2' p_2(x) - \pi_1 C_1' p_1(x)]\,dx \tag{2.1}$$

Consider the following quantity which may be regarded as generalization of Kolmogorov's measure of variational distance:

$$\begin{aligned} K(\pi) &= \frac{1}{2}\int_X |\pi_1 C_1' p_1(x) - \pi_2 C_2' p_2(x)|\,dx \\ &= \frac{1}{2}\pi_1(C_{11}+C_{12}) + \frac{1}{2}\pi_2(C_{22}+C_{21}) - R(\pi) \end{aligned} \tag{2.2}$$

An upper bound of $K(\pi)$ is obtained from the Cauchy-Schwartz inequality as follows:

$$K(\pi) \le \frac{1}{2}[(\pi_1 C_1' + \pi_2 C_2')^2 - 4\pi_1\pi_2 C_1' C_2' \rho^2]^{1/2} \tag{2.3}$$

A lower bound of $K(\pi)$ is

$$K(\pi) \ge \frac{1}{2}[\pi_1 C_1' + \pi_2 C_2' - 2\sqrt{\pi_1\pi_2 C_1' C_2'}\,\rho] \tag{2.4}$$

From Eqs. (2.2) - (2.4) an upper and lower-bound of the Bayes risk is given by

$$R_0(\pi) + \frac{2\pi_1\pi_2 C_1' C_2' \rho^2}{\pi_1 C_1' + \pi_2 C_2' + \sqrt{(\pi_1 C_1' + \pi_2 C_2')^2 - 4\pi_1\pi_2 C_1' C_2' \rho^2}} \le R(\pi) \le R_0(\pi) + \sqrt{\pi_1\pi_2 C_1' C_2'}\,\rho \tag{2.5}$$

where $R_0(\pi) = \pi_1 C_{11} + \pi_2 C_{22}$.

Let us adopt a simpler but less strict lower bound:

$$R_0(\pi) + \frac{\pi_1\pi_2 C_1' C_2'}{\pi_1 C_1' + \pi_2 C_2'}\cdot\rho^2 \le R(\pi) \le R_0(\pi) + \sqrt{\pi_1\pi_2 C_1' C_2'}\,\rho \tag{2.6}$$

When $C_{11} = C_{22} = 0$, $C_{12} = C_{21} = 1$, the average loss is equal to $P_e(\pi)$, the error probability, and Eqs. (2.5) and (2.6) become

$$\frac{2\pi_1\pi_2\rho^2}{1 - \sqrt{1 - 4\pi_1\pi_2\rho^2}} \le P_e(\pi) \le \sqrt{\pi_1\pi_2}\cdot\rho \tag{2.8}$$

and

$$\pi_1\pi_2\rho^2 \le P_e(\pi) \le \sqrt{\pi_1\pi_2}\cdot\rho \tag{2.8}$$

Thus we see that Eq. (2.5) - (2.9) make the minimization of ρ a reasonable criterion of optimality. Minimization of ρ is equivalent to maximization of the B-distance defined by Eq. (1.8).

Another distance measure frequently used is the J-divergence of Eq. (1.5). Unlike the B-distance, no upper bound on $P_e(\pi)$ in terms of J has been given[1], although a lower bound is

$$P_e(\pi) \geq \pi_1\pi_2\ \rho^2 > \pi_1\pi_2\ \exp(-J/2) \tag{2.9}$$

Because of this fact, Kailath [2] claims that the B-distance is a better criterion for signal selection problems than J-divergence. Another superiority of the B-distance is shown in cases where the average with respect to *unknown parameters* is required, such as in non-coherent reception problems. The next section is intended to demonstrate this fact.

3. APPLICATION OF THE B-DISTANCE TO COMMUNICATION PROBLEMS

Example 3.1 Detection of a Signal with Unknown Phase and Amplitude

Let the observable be the continuous signal x(t) such that

$$\text{under } H_1 \quad : \quad x(t) = n(t) \tag{3.1}$$

$$\text{under } H_2 \quad : \quad x(t) = \alpha s(t) + n(t) \tag{3.2}$$

where s(t) is the complex envelope of a known signal and α is a complex Gaussian variable [9] with probability distribution

$$p(\alpha)\,d\alpha d\alpha^* = \frac{1}{2\pi D_\alpha^2}\ \exp\left(-\frac{\alpha\ \alpha^*}{D_\alpha^2}\right) d\alpha d\alpha^* \tag{3.3}$$

The process n(t) is the complex envelope of a stationary Gaussian noise process, with zero mean and covariance

$$E[n(s)n^*(t)] = R(s-t) \tag{3.4}$$

The B-coefficient between H_1 and H_2, given α, can be calculated as

$$\rho(\alpha) = \exp\left(-\frac{1}{4}\| s(t) \|^2_{H(R)}\ \alpha\ \alpha^*\right) \tag{3.5}$$

where $\| s(t) \|^2_{H(R)}$ is the norm square of s(t) on the reproducing kernel Hilbert space generated by the covariance function R(s-t). On taking the expectation of Eq. (3.5) with respect to α, we have

$$E_\alpha\{\rho(\alpha)\} = \frac{1}{1+\frac{1}{4}\mu} \tag{3.6}$$

where

$$\mu = \|s(t)\|^2_{H(R)} \cdot D_\alpha^2$$

Here the quantity μ may be regarded as the S/N ratio. Then the B-distance is

$$B = \ln(1+\frac{1}{4}\mu) \tag{3.7}$$

This result can be applied readily to a signal selection problem in the same way as Grettenberg [1] applied the J-divergence to the detection of a completely known signal [8].

---

[1] For the particular case of selection between two Gaussian probability measures, we have [3]

$$\sqrt{\pi_1\pi_2}\ \exp\{(J/2)^{-1/4}\} > P_e(\pi) \geq \pi_1\pi_2\ \exp(-J/4)$$

Example 3.2 Radar Parameters Estimation

The following problem was originally studied by Grettenberg[1] by the J-divergence method. Let τ and λ be the time delay and doppler-shift frequency of the returned echo. The τ-λ plane can be divided into a set of resolution cells, which Grettenberg calls a message space M:

$$M = \{m = (\tau_m, \lambda_m)\} \tag{3.8}$$

Let w(t) be the complex envelope of the transmitted signal. Assuming that the amplitude is known and is common to all cells m in the range of interest, the observed data under the hypothesis H_m is given by

$$x(t) = e^{j\theta_m} w(t-\tau_m) e^{j2\pi\lambda_m t} + n(t) = e^{j\theta_m} w_m(t) + n(t) \tag{3.9}$$

The phase θ_m is not measured by the observer and hence is assumed to be a random variable, uniformly distributed between 0 and 2π.

The B-coefficient between H_m and H_q $(m,q \in M)$ is given after some manipulation [8] as follows:

$$\rho(H_m,H_q) = \exp\left[-\frac{1}{4} \|w_m(t) - w_q(t)\|^2_{H(R)}\right] \tag{3.10}$$

The reproducing kernel inner product which appears in Eq. (3.10) can be written in terms of the generalized ambiguity function as follows:

$$\begin{aligned} [w_m(t), w_q(t)]_{H(R)} &= [w(t-\tau_m) e^{j2\pi\lambda_m t}, w(t-\tau_q) e^{j2\pi\lambda_q t}]_{H(R)} \cdot e^{-j(\theta_m-\theta_q)} \\ &= \chi_w(\tau_m-\tau_q, \lambda_m-\lambda_q)\, e^{-j\theta_{mq}} \end{aligned} \tag{3.11}$$

where

$$\theta_{mq} = \theta_m - \theta_q \tag{3.12}$$

Then Eq. (3.10) can be written as

$$\rho(H_m,H_q) = \exp\{-\frac{1}{2} [\chi_w(0,0) - |\chi_w(\tau_m-\tau_q, \lambda_m-\lambda_q)| \cos\theta]\} \tag{3.13}$$

$$\text{where } \theta = \theta_{mq} - \arg.\ \chi_w(\tau_m-\tau_q, \lambda_m-\lambda_q) \tag{3.14}$$

Now the averaged B-coefficient is given by

$$E_\theta\{\rho(H_m,H_q)\} = \exp\{-\frac{1}{2} \chi_w(0,0)\} I_0[\frac{1}{2}|\chi_w(\tau_m-\tau_q, \lambda_m-\lambda_q)|] \tag{3.15}$$

Therefore the B-distance is

$$B(H_m,H_q) = \frac{1}{2} \chi_w(0,0) - \ln I_0[\frac{1}{2}|\chi_w(\tau_m-\tau_q, \lambda_m-\lambda_q)|] \tag{3.16}$$

The maximization of the B-distance is equivalent to the minimization of the absolute value of the ambiguity function; this latter approach is well known in radar signal design.

4. DISTANCE MEASURES AND ASYMPTOTIC RELATIVE EFFICIENCY

Thus far we have considered the distance between two probability measures defined on the space of the observed data x and the relation of this distance to performance criterion such as the Bayes risk and error probability. Let us turn our attention from the observed raw data to condensed data, i.e., to a statistic T(x). Estimation and detection procedures can be regarded as data reduction processing and can enable us to discriminate between alternative situations.

For a given statistic T(x), the amount of discrimination between H_1 and H_2 provided by T(x) will be defined by the distance B or J:

$$B_T = -\ln\rho_T \quad , \quad \text{where } \rho_T = \int \sqrt{p(T/H_1)\,p(T/H_2)}\; dT \qquad (4.1)$$

and

$$J_T = E_1[\ln \frac{p(T/H_1)}{p(T/H_2)}\;] + E_2[\ln \frac{p(T/H_2)}{p(T/H_1)}] \qquad (4.2)$$

It can be shown [4] that each of these measures is no larger than the corresponding expression when T is replaced by the original data x, i.e., $B_T \le B_x$ and $J_T \le J_x$. The equality in these relations holds if and only if T(x) is a sufficient statistic. Therefore the ratio of B_T to B_x may represent the efficiency of T(x), and the same thing can be said about the J-divergence.

Let T_α and T_β be any two consistent test statistics for testing H_1: $\theta = \theta_o$ against H_2: $\theta > \theta_o$ (or $\theta < \theta_o$). Let the data of T_α and T_β consist of n_α and n_β independent observations, respectively. The most frequently used criterion for comparing the efficiency of two statistics is the A.R.E. (Asymptotic Relative Efficiency) defined [10] by

$$A_{\beta\alpha} = \lim_{\theta\to\theta_o} \frac{n_\alpha}{n_\beta} \qquad (4.5)$$

where the limiting operation is taken under the constraint that the errors of type I and II of T_α and T_β are each kept equal.

The smaller the distance, the more difficult it will be to discriminate H_2 from H_1, and hence the larger the amount of data that will be required to achieve the assigned performance. Therefore it is clear that some relationship exists between A.R.E. and the distance measures. Assume that T_α is asymptotically normally distributed whatever the value of θ may be, with mean $M_\alpha(\theta)$ and variance $D_\alpha^2(\theta)$, $\theta \ge \theta_o$. Assume that the first (m-1)st order derivatives of $M_\alpha(\theta)$ are zero at $\theta = \theta_o$, i.e., $M_\alpha^{(i)}(\theta_o) = 0$ for i = 1,2,...,m-1 and $M_\alpha^{(m)}(\theta_o) \neq 0$. Then it can be shown [8] after some manipulation that the B-coefficient between $\theta = \theta_o$ and $\theta = \theta_o + \delta\theta$ is given by

$$\rho_{T_\alpha}(\theta_o, \theta_o+\delta\theta) = 1 - \frac{1}{8}\left(\frac{\delta\theta^m}{m!}\right)^2 \cdot I_{\alpha,m} + O(\delta\theta^{3m}) \qquad (4.6)$$

where

$$I_{\alpha,m} = E\left(\frac{\partial^m \ln p(T_\alpha/\theta_o)}{\partial\theta^m}\right)^2 = \left[\frac{M_\alpha^{(m)}(\theta_o)}{D_\alpha(\theta_o)}\right]^2 \qquad (4.7)$$

Therefore the B-distance provided by the statistic $T_\alpha(x)$ is

$$B_{T_\alpha}(\theta_o,\theta_o+\delta\theta) = \frac{1}{8}\left(\frac{\delta\theta^m}{m!}\right)^2 \cdot I_{\alpha,m} + O(\delta\theta^{3m}) \qquad (4.8)$$

For the case m = 1, the quantity $I_{\alpha,m}$ is equal to Fisher's information measure; hence $I_{\alpha,m}$ may be called a generalized Fisher's information measure. On assuming that the asymptotic distribution of T_β behaves essentially in the same fashion as T_α, it follows that

$$B_{T_\beta} / B_{T_\alpha} = \left[\frac{M_\beta^{(m)}(\theta_o)}{D_\beta(\theta_o)}\right]^2 \Big/ \left[\frac{M_\alpha^{(m)}(\theta_o)}{D_\alpha(\theta_o)}\right]^2 \qquad (4.9)$$

Let $\frac{M_\alpha^{(m)}(\theta_o)}{D_\alpha(\theta_o)} \sim c_\alpha \cdot n_\alpha^{m\cdot r}$ as $n_\alpha \to \infty$. This relation defines the constants c_α and r. We define the efficacy of T_α by

$$E_\alpha = \lim_{n_\alpha \to \infty} \left[\frac{M_\alpha^{(m)}(\theta_o)}{n_\alpha^{mr}\; m!\; D_\alpha(\theta_o)}\right]^{\frac{1}{m\cdot r}} < \infty \qquad (4.10)$$

The efficacy defined here is a generalization of the term given by Capon [11]. Then, keeping $n_\alpha = n_\beta$, we have the following asymptotic relation:

$$\lim_{\delta\theta \to 0} \frac{B_{T_\beta}(\theta_o,\theta_o+\delta\theta)}{B_{T_\alpha}(\theta_o,\theta_o+\delta\theta)} = {\frac{E_\beta}{E_\alpha}}^{2m\cdot r} = A_{\beta\alpha}^{2m\cdot r} \qquad (4.11)$$

If m = 1 and $r = \frac{1}{2}$, as is the case for many problems, then $\frac{B_{T_\beta}}{B_{T_\alpha}} \to A_{\beta\alpha}$.

Similar results can be obtained for the J-divergence, i.e.,

$$J_{T_\alpha}(\theta_o,\theta_o+\delta\theta) = \left(\frac{\delta\theta^m}{m!}\right)^2 \cdot I_{\alpha,m} + O(\delta\theta^{3m}) \qquad (4.12)$$

and hence

$$J_{T_\alpha}(\theta_o,\theta_o+\delta\theta) = 8\, B_{T_\alpha}(\theta_o,\theta_o+\delta\theta) \quad \text{as } \delta\theta \to 0 \qquad (4.13)$$

Therefore it follows that

$$B_{T_\beta} / B_{T_\alpha} = J_{T_\beta}/J_{T_\alpha} \to A_{\beta\alpha}^{2m\cdot r} \qquad (4.14)$$

5. THE DETECTION PROBLEM IN TWO-INPUT SYSTEMS

Let us consider a detection problem in a two-input system, where the stochastic signal s(i) is common to both channels, say "channel a" and "channel b". Noises are additive and independent, i.e.,

$$\begin{aligned} x_a(i) &= \sqrt{\theta}\; s(t) + n_a(t) \\ x_b(i) &= \sqrt{\theta}\; s(t) + n_b(t) \end{aligned} \quad , \quad i = 1,2,\ldots,N \qquad (5.1)$$

where s(i), $n_a(i)$ and $n_b(i)$ are real Gaussian random processes, independent of each other and with zero mean and correlation $E[s(i)s(j)] = P_s(i)\ \delta_{ij}$, $E[n_a(i)n_a(j)] = E[n_b(i)n_b(j)] = P_n(i)\delta_{ij}$

Here θ is the signal-to-noise ratio parameter. Consider the testing hypothesis problem: $\theta = 0$ against $\theta > 0$. The B-coefficient between the two hypotheses, based on the observed data $x_a(i)$ and $x_b(i)$, $i = 1,2,\ldots,N$ is given by

$$\rho_x(0,\theta) = \prod_{i=1}^{N} [1 + 2\theta\cdot\mu(i)]^{1/4}\ [1 + \theta\cdot\mu(i)]^{-1/2} \qquad (5.2)$$

where $\mu(i) = P_s(i)/P_n(i)$. Therefore the B-distance is given by

$$B_x(0,\theta) = \frac{1}{2}\sum_{i=1}^{N} \left[\ln(1+\theta\mu(i)) - \frac{1}{2}\ln(1+2\theta\mu(i))\right] \qquad (5.3)$$

Now consider the case where the data $x_a(i)$ and $x_b(i)$ are passed through hard limiters, the output of which will be denoted by $y_a(i)$ and $y_b(i)$:

$$y_a(i) = \operatorname{sgn} x_a(i) \quad \text{and}\ y_b(i) = \operatorname{sgn} x_b(i)\ ,\ i = 1,2,\ldots,N$$

The B-coefficient and the B-distance provided by $y_a(i)$ and $y_b(i)$ are given, after some manipulation [8], by

$$\rho_y(0,\theta) = \prod_{i=1}^{N}\frac{1}{2}\left\{\sqrt{1+\frac{2}{\pi}\sin^{-1}\frac{\theta\cdot\mu(i)}{1+\theta\cdot\mu(i)}} + \sqrt{1-\frac{2}{\pi}\sin^{-1}\frac{\theta\cdot\mu(i)}{1+\theta\cdot\mu(i)}}\right\} \qquad (5.4)$$

and

$$B_y(0,\theta) = -\ln \rho_y(0,\theta) \qquad (5.5)$$

Now consider another reduction of the data obtained by defining a new random process z(i) as the product of the two channels, i.e.,

$$z(i) = x_a(i)\cdot x_b(i)\ ,\quad i = 1,2,\ldots,N \qquad (5.6)$$

The distribution density of z(i) is given, after some manipulation [8], by

$$p(z(i)/\theta) = \frac{1}{\pi P_n(i)\sqrt{1+2\theta\mu(i)}}\exp\left\{\frac{\theta\mu(i)z(i)}{(1+2\theta\mu(i))\cdot P_n(i)}\right\} \times \qquad (5.7)$$

$$K_o\left(\frac{(1+\theta\mu(i))\,|z(i)|}{(1+2\theta\mu(i))\cdot P_n(i)}\right)$$

where $K_0(\cdot)$ is the modified Bessel function of the second kind and of order zero. The B-coefficient based on z(i), $i = 1,2,\ldots,N$ can be calculated for the weak signal case, i.e., for $\theta \ll 1$, as follows:

$$\rho_z(0,\theta) = \sum_{i=1}^{N}(1+2\theta\mu(i))^{-1/4}\left[\frac{1+\theta\mu(i)}{1+2\theta\mu(i)} - \left\{\frac{\theta\mu(i)}{2(1+2\theta\mu(i))}\right\}^2\right]^{-1/2} \qquad (5.8)$$

and

$$B_z(0,\theta) = -\ln \rho_z(0,\theta) \tag{5.9}$$

Let $\theta << 1$ in Eqs. (5.3), (5.5) and (5.9); then we will have

$$B_x(0,\theta) = \frac{\theta^2}{4} \sum_{i=1}^{N} \mu^2(i) \tag{5.10}$$

$$B_y(0,\theta) = \frac{\theta^2}{2\pi^2} \sum_{i=1}^{N} \mu^2(i) \tag{5.11}$$

$$B_z(0,\theta) = \frac{\theta^2}{8} \sum_{i=1}^{N} \mu^2(i) \tag{5.12}$$

Taking the ratio of these distances, we obtain $B_y/B_x = \frac{1}{2}(\frac{2}{\pi})^2$ and $B_z/B_x = \frac{1}{2}$. These quantities are equal, respectively, to the A.R.E.'s of the Polarity Coincidence Correlator (P.C.C.) and the Correlator, with respect to the optimum detector[12]. This is the case since the optimum detector T_{opt}, the Polarity Coincidence Correlator T_{pcc}, and the Correlator T_c are sufficient test statistics (for $\theta << 1$) of x, y, and z, respectively, for distinguishing between the two hypotheses. These systems are given by

$$T_{opt} = \sum_{i=1}^{N} \mu(i) \cdot \{x_a(i) + x_b(i)\}^2 \tag{5.13}$$

$$T_{pcc} = \sum_{i=1}^{N} \mu(i) \cdot y_a(i) \cdot y_b(i) \tag{5.14}$$

$$T_c = \sum_{i=1}^{N} \mu(i) \cdot z(i) \tag{5.15}$$

ACKNOWLEDGMENT

This work was supported by the National Science Foundation under Grants GK-187 and GK-1439 and by the U.S. Army Research Office - Durham, under Contract DA-31-124-ARO-D-292.

REFERENCES

1. T. L. Grettenberg, "Signal Selection in Communication and Radar Systems", IEEE Trans. on Information Theory, Vol. IT-9, pp. 265-275; October 1963.
2. T. Kailath, "The Divergence and Bhattacharyya Distance Measures in Signal Selection", IEEE Trans. on Communication Technology, Vol. COM-15, pp. 52-60; February 1967.
3. T. T. Kadota and L. A. Shepp, "On the Best Finite Set of Linear Observables for Discriminating Two Gaussian Signals", IEEE Trans. on Information Theory, Vol. IT-13, pp. 278-284; April 1967.
4. S. M. Ali and S. D. Silvey, "A General Class of Coefficients of Divergence of One Distribution from Another", J. Royal Stat., Vol. 28, pp. 131-142; 1966.
5. S. Kullback, Information Theory and Statistics, New York,

Wiley; 1959.

6. K. Matsushita, "A Distance and Related Statistics in Multivariate Analysis", in P. R. Krishnaiah, ed., Multi-variate Analysis, pp. 187-200, Academic Press; 1966.

7. S. Karlin and R. N. Bradt, "On the Design and Comparison of Dichotomous Experiments", Ann. Math. Stat., Vol 27, pp. 390-409; 1956.

8. H. Kobayashi, "Distance Measures and Related Criteria", Chapter V of Representations of Complex-Valued Vector Processes and Their Application to Estimation and Detection, Ph.D. Thesis, Princeton University, N.J.; August 1967.

9. Ibid., Chapter I, "A Representation of the Complex Gaussian Distribution and Its Application to Complex Gaussian Processes".

10. M. G. Kendall and A Stuart, The Advanced Theory of Statistics, Vol. 2, Inference and Relationship, Hafner Publishing Co., New York.

11. J. Capon, "On the Asymptotic Efficiency of Locally Optimum Detectors", IRE Trans. on Information Theory, Vol. IT-7, pp. 67-71; April 1961.

12. J. W. Carlyle and J. B. Thomas, "On Nonparametric Signal Detectors", IEEE Trans. on Information Theory, Vol. IT-10, pp. 146-152; 1964.

SIGNAL DESIGN FROM A VARIATIONAL VIEWPOINT

R. L. Kashyap
Purdue University
Lafayette, Indiana

It is desired to transmit the value of a parameter θ over a discrete additive white noise channel over a time t_f using a noiseless feedback channel. Let $N(t_f)$ iterations be performed in time t_f. The coding scheme can be described as follows. At discrete instants $i = 0,1,\ldots,N-1$ the transmitter sends out signals $v(i)$. These will be received as measurements $y(i)$, $i = 0,1,\ldots$ by the receiver after being corrupted by white noise η of covariance r.

$$y(i) = v(i) + \eta(i)$$

At the i-th stage the receiver constructs the estimate $x(i)$ of θ based on the available measurements and sends it back to the transmitter over the noiseless feedback link. The problems are as follows:

A) Determine the signals $v(i)$, $i = 0,1,\ldots,N-1$ and the estimates $x(i)$, $i = 1,\ldots,N$ to minimize the conditional mean square error $E[(x(N)-\theta)^2/y(0),\ldots,y(N-1)]$ subject to the constraints mentioned below.

(i) Average Power: $P_{AV} = E(\sum_{i=0}^{N-1} v_i^2)/t_f$

(ii) Linear Transmitter: $v(i) = K_2(i)(\theta - x(i))$

(iii) Gain $|K_2(i)| \leq \beta$

B) Assuming that the parameter θ can assume only one of finite number M of values, choose a decision rule at the receiver for decoding the message based on $x(N)$. Compute the corresponding probability of error and study its asymptotic behavior as $t_f \to \infty$

The optimization problem can be solved in closed form. The results of the study serve to isolate the optimal coding schemes from the non-optimal ones. The results of the paper can be extended to cover channels with correlated noise, multiplicative noise and time continuous channels.

The work was partially supported by NSF under Grant GK-1970.

ADAPTIVE RADAR DETECTION WITH REGULATED ERROR PROBABILITIES IN EXTENDED CLUTTER TARGET ENVIRONMENTS

HAROLD M. FINN
Radio Corporation of America, Defense Electronic Products
Moorestown, New Jersey 08057

ABSTRACT

Sequential detection modes designed for employment in statistically non-stationary extended clutter target environments are described. The statistical hypothesis tests developed are adaptive in the sense that spatially sampled maximum likelihood estimates of the clutter target scattering function, made on the first sequential step control the clutter discrimination policy of the second step. In addition, updated estimates of these quantities are employed to control the second step detection thresholds. The adaptive modes are shown to yield a false alarm probability invariant to changes in the clutter noise power density, and a detection probability which is greater than a specified lower bound for an hypothesized target and range of clutter levels.

INTRODUCTION

The statistical hypothesis test performed at each beam position of the surveillance volume and involved in the radar detection of a return signal of generally known signal waveform characteristics immersed in an unknown extended clutter target environment is the subject of this note.

The formulation of the radar detection problem for the case where the transmitted signal is S (t) involves as the observed random variables $\{Y_i\}$ for each transmission, the magnitude of the cross correlations of the received signal V (t) with delayed τ and doppler shifted f replicas of the signal $\{S(t;\ \tau_i,\ f_i)\}$ or $Y_i = \left| \int S\ (t;\ \tau_i\ ,\ f_i\ )\ V(t)\ dt \right|$, or equivalently the sampled and linearly detected matched filter outputs. The receiver input when a target is present is $V\ (t) = a\ s(t;\ \tau,\ f) + C\ (t) + m\ (t)$ where C (t) is the clutter noise, the return of the spatially distributed collection of randomly oriented point targets, and m (t) is the additive receiver noise with noise power density No. The customary idealization employed in this development in order to separate the signal detection from the signal parameter estimation problem involves assuming that the target can appear at discrete points in the admissible range delay τ - doppler shift f target parameter space. For the transmission of a signal of bandwidth W and time duration T, this involves dividing the admissible target parameter space into resolution cells separated in range delay by the range resolution constant $\frac{1}{\ }$, and in the doppler dimension by the doppler resolution constant of $\frac{1}{T}$,W and assuming that the signal return from a target can only occur at the center of one of these resolution cells. N_F resolution cells cover the admissible target parameter space and the 'output' of the ith resolution cell is the observed random variable Y_i .

The clutter 'cloud' is assumed to cover the target parameter space of interest and be of the spatially incoherent type. That is to say, returns from disjoint portions of the parameter space are uncorrelated. For this clutter model, the variance of the output Y_i of the ith resolution cell, σ_i^2 may be expressed as:

$$\sigma_i^2 = \int\!\!\int \left| X\ (\tau,\ f) \right|^2\ \rho(\ \tau + \tau_i\ ,\ f + f_i\ )\ d\tau\ df + No$$

the two dimensional convolution of the Woodward signal ambiguity$|X(\tau, f)|^2$ ($X(\tau, f) = \int U(t)\, U(t - \tau) e^{-J2\pi f t}\, dt$ and the clutter target scattering function $\rho(\tau, f)\, d\tau\, df$.(U (t) is the complex transmitted modulation function). $\rho(\tau, f)\, d\tau\, df$ represents the average value of the energy returned from the clutter targets in the elementary region having a range of delay between τ and $\tau + d\tau$ and a doppler shift between f and f + df.(1)

A fixed sample size test for the detection involves comparing each of the cell outputs (Y_i) after a transmission with a fixed threshold, and accepting the target present hypothesis (H_1) if a threshold crossing occurs in at least one of these cells. The extreme sensitivity of the single cell false alarm probability P_{FA} of such a fixed sample size test is indicated in Figure 1A where it is noted that a 3 db increase in the total noise power density (clutter plus thermal noise) causes a 10,000 fold increase in the false alarm probability (a change from the design value of $P_{FA} = 10^{-8}$ to $P_{FA} = 10^{-4}$). A dual aspect of this detection problem becomes apparent if one considers a system, where in some manner the detection threshold can be raised and reset to compensate for the 3 db increase in noise level so that the false alarm probability is again set at $P_{FA} = 10^{-8}$. The detection probability (P_D) for this case is noted in Figure 1B to fall from the design value of $P_D = 0.9$ to $P_D = 0.15$.

The adaptive detection procedures described in this note achieve a false alarm probability invariant with clutter level changes, and a lower bound of the detection probability over a specified range of clutter level changes and for a hypothesized target. (The desired characteristics are also presented in Figures 1A and 1B.) The cost for performing the test which is assumed to be proportional to the relative sophistication of the discriminant employed on the second sequential step is made a monotonically increasing function of the encountered clutter levels. Consequently, in a non-uniform clutter environment (see Figure 2 for an hypothetical case) the test cost of the adaptive mode changes with the encountered threat, so that the average cost is less than that required for a non-adaptive fixed sample test which must be designed to handle a 'worst-case' clutter at each beam position.

ADAPTIVE DETECTION MODE FORMULATION

Two adaptive mechanisms are embedded within a decision processor framework with two sequential steps in order to achieve the regulation of the error probabilities. The mechanism for bounding the detection probability involves making the degree of clutter discrimination employed on the second step, a monotonically increasing function of the spatially sampled clutter level estimates obtained on the first step; and the false alarm probability regulation mechanism involves the use of detection thresholds made proportional to updated estimates of these levels on the second sequential step.

Target Scattering Function Estimates

The second step discriminant is controlled by the first step $\hat{\phi}_{max} = \mathrm{Max}\{\hat{\phi}_i\}$ where ϕ_i is the ratio of the variance of the ith cell output in the clutter plus noise environment(σ_i^2) to the output variance in the thermal noise alone case (No) or $\hat{\phi}_i = \frac{\sigma_i^2}{N}$. The decision statistics formed at the output of the second step are then compared with thresholds made proportional to updated estimates of the set of $\{\sigma_i^2\}$.

The estimate $\hat{\sigma}_i{}^2$ may be expressed as:

$$\hat{\sigma}_i{}^2 = \iint |x(\tau, f)|^2 \hat{\rho}(\tau + \tau_i, f + f_i)\, d\tau\, df + No$$

where the thermal noise power density term is assumed to be a known quantity. The spatially sampled maximum likelihood estimation procedure of the clutter target scattering function $\rho(\tau, f)\, d\ \ df$ is a key element of these detection modes since it is involved in both adaptive mechanisms. For the spatially non-coherent extended clutter phenomena, the estimate $\hat{\rho}(\tau, f)\, d\tau\, df$ at a resolution cell can be obtained on a single transmission by employing the N_c resolution cells surrounding the cell under test. This is a basic point - the search for a signal return after a transmission is a multiple resolution cell operation, and in this procedure these multiple resolution cell outputs embraced by a common extended clutter cloud are also employed to obtain an estimate of the clutter target scattering function $\rho(\tau, f)\, d\tau\, df$. The target scattering function to be estimated is assumed to be of the form:

$$\rho(\tau, f)\, d\tau\, df = C_k\, \rho_k'(\tau, f)\, d\tau\, df \text{ for } K = 1, 2, \text{---} G.$$

This assumed form implies that $\rho(\tau\ f)\, d\tau\, df$ may be expressed as the product of an unknown magnitude C_k and a target scattering function 'shape-factor' $\hat{\rho}_k'(\tau, f)\, d\tau\, df$ which is one of G admissible forms. The estimator $\hat{\rho}(\tau, f)\, d\tau\, df$ is the expression maximizing the likelihood $L(x_1, x_2 \text{---} x_{N_c} \mid C_k\, \rho_k'(\tau, f)\, d\tau\, df)$ where this likelihood L is the expression of the conditional probability of the observed resolution cell outputs $\{x_1, x_2, \text{---}\ x_{N_c}\}$ surrounding the cell under test on the hypothesis that the target scattering function $\rho(\tau, f)\, d\tau\, df = C_k\, \rho_k'(\tau, f)\, d\tau\, df$. For the assumed spatially non-coherent extended clutter model where the number of scatterers in each resolution cell is sufficiently large for an application of the central limit theorem, the resolution cell outputs are approximately raleigh distributed and statistically independent, and the likelihood for this case may be expressed as:

$$L(x_1, x_2, \text{---} x_{N_c} \mid C_k\, \rho_k'(\tau, f)\, d\tau\, df) = \prod_{i=1}^{N_c} \frac{X_i}{\sigma_{i(k)}^2}\, e^{-\frac{X_i^2}{2\sigma_{i(k)}^2}}\, dX$$

where $\sigma_{i(k)}^2 = \frac{1}{2}(C_k \gamma_{i(k)} + N_o)$ and $\gamma_{i(k)}^2 = \iint |X(\tau, f)|^2 \rho_K'(\tau + \tau_i, f + f_i)\, d\tau\, df$.

The maximum likelihood estimate $\hat{\rho}(\tau, f)\, d\tau\, df$ maximizing L for the practical clutter noise limiting environment (i.e., where the clutter power density $\iint |X(\tau, f)|^2\ \rho(\tau + \tau_o, f + f_o)\, d\tau\, df$ is much larger than the thermal noise power density N_o) may be expressed as

$$\hat{\rho}(\tau, f)\, d\tau\, df = \hat{C}_\ell\, \hat{\rho}_\ell'(\tau, f)\, d\tau\, df$$

where $$\hat{C}_\ell = \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{X_i^2}{\gamma_{i(\ell)}}$$

and ℓ is any one of the G target scattering function 'shape factors'

which maximizes the quantity $$L_k' = l_n \left[\frac{1}{\hat{C}_k \sum_{i=1}^{N_c} \gamma_{i(k)}} \right] \text{ or } \left(L_\ell' = \text{Max}(L_k') \right)$$

Employing the expression of the ensemble average of the square of the matched filter outputs σ_o^2 for the clutter limiting environment where the thermal noise power density may be neglected,

$$\hat{\sigma}_o = \frac{1}{N_c}\sum_{i=1}^{N_c}\frac{x_i^2}{W_i(\ell)} \quad \text{where} \quad W_i(\ell) = \frac{\gamma_{i(\ell)}^2}{\gamma_o^2}$$

The estimate $\hat{\sigma}_o^2$ is noted to be a weighted sum of the N_c square law detector outputs surrounding the cell under test, an operation which is practical to implement for each of the resolution cells under test. The adaptive mode probability distributions which are developed in this note are based on a simpler assumed distribution - namely, that the shape factor of the target scattering function $\rho'(\tau, f)\, d\tau\, df$ is a priori information and the magnitude C only is estimated. (That is

$$\rho(\tau, f)\, d\tau\, df = C\rho'(\tau, f)\, d\tau\, df$$

and for this case the estimator $\hat{\sigma}_o^2 = \frac{1}{N_c}\sum_{i=1}^{N_c}\frac{x_i^2}{W_i}$.

Variable Non-Coherent Integration Gain Detection Mode

Two adaptive detection modes are developed in the note and they differ from one another in the adaptive mechanism used for regulating the detection probability. For both detection modes, there are M possible quantized second-step discrimination policies arranged in the order of increasing cost and effectiveness in clutter return discrimination. The Jth discrimination policy is employed whenever the estimate $\hat{\phi}_{max}$ lies between the second step discrimination policy making thresholds ϕ_{T_J} and $\phi_{T_{J-1}}$.

That is for $\phi_{T_{J-1}} < \hat{\phi}_{Max} < \phi_{T_J} \Longrightarrow$ Jth policy, and for $\phi_{T_{M-1}} \leq \hat{\phi}_{max} \Longrightarrow$ Mth policy

where M is the maximum permissible discrimination policy at a beam position. This latter policy rule places a limit on the sensor system expenditure at a beam position; i.e., regardless of the intensity of the encountered clutter, the expenditure never exceeds a value corresponding to the maximum second step discriminant M, so that a degradation of performance results for encountered clutter levels exceeding a specified value. One of the cases considered involves a control of the number of second sequential step transmissions as a function of the first step estimate $\hat{\phi}_{max}$, and where each of these transmissions is designed to yield statistically independent clutter returns. The total number of transmissions employed is J corresponding to the Jth policy. (The number employed on the second sequential step is J - N_1, where N_1 is the deterministic number employed on the first step. The decision statistic Z_i for this case is the non-coherent sum of the J square law detector outputs $\{\zeta_{Ki}\}$ at the ith resolution cell and may be expressed as

$$Z_i = \sum_{K=1}^{J}\zeta_{Ki} \qquad \left(\zeta_{Ki} = \frac{x_{Ki}^2}{2}\right)$$

Since each of these transmissions is designed to yield a statistically independent clutter return, a controllable non-coherent integration gain and discrimination against the clutter results. The method for achieving the statistical independence of the clutter returns of the J transmissions may be accomplished by employing one or more of the following measures: 1) the interval between transmissions may be made long with respect to the correlation time of the clutter, 2) a sufficient carrier-frequency 'jump' may be employed on each transmission to effect the statistical independence of the sum of the field vectors representing the returns from the discrete scatterers in a resolution cell, and 3) the waveform design may be changed for each transmission in such a manner so that the intersection, or convolution, of the clutter target scattering function $\rho(\tau, f)\, d\tau\, df$ and the waveform signal ambiguity function $|X(\tau, f)|^2$ involvesdifferent and statistically independent contributions of the extended clutter phenomena. The final decision rule for this detection mode involves comparing each of the decision statistics $Z_i$with a threshold $T_J^{(i)}$ made proportional to a second sequential step maximum likelihood estimate of the output variance of the cell or

$$T_J^{(i)} = K_J\, \hat{\sigma}_i^{\ 2}.$$

(K_J is the threshold proportionality constant used when the Jth policy is employed.) The target present hypothesis (H_1) is accepted if at least one threshold crossing event occurs. That is the decision rule is to accept the H_1 hypothesis if

$$\bigcup_{i=1}^{N_F} (E: Z_i > T_J^{\,i})$$

is true, and otherwise to reject this hypothesis. The threshold control procedure is the false alarm probability regulation mechanism and with its employment, the false alarm probability is maintained constant and invariant to changes in the total noise power density.[2] This adaptive mode will be referred to as the variable non-coherent integration gain detection mode. See Figure 3 for a functional diagram.

Resolution-Variant Detection Mode

A resolution-variant detection mode is the second of the adaptive statistical hypothesis tests treated in this note. The adaptive mechanism employed in this mode for regulating the detection probability involves making the second sequential step transmitted signal time duration-bandwidth product (T_2W_2) a monotonically increasing function of the estimated clutter level $\hat{\phi}_{max}$. The effective clutter power density after matched filter processing is assumed to be inversely proportional to the transmitted signal time duration bandwidth TW for the radar cases considered, and the expenditure of a 'look' at a beam position is assumed to be proportional to the ratio of the signal time duration product of the second sequential step to that employed on the first step $\frac{T_2W_2}{T_1W_1}$. For the analyzed case, the signal time duration bandwidth product is constrainted to increases of octave steps, and the Jth second step discrimination policy involves making the ratio $\frac{T_2W_2}{T_1W_1} = 2^{J-1}$. The first discrimination policy ($J = 1$) involves no change in the transmitted signal time duration bandwidth product from the first to the second step and the maximum discrimination policy (M) involves making $\frac{T_2W_2}{T_1W_1} = 2^{M-1}$. The decision statistic (Z_i)formed at the output of each of the second step resolution cells is $Z = \zeta'_{i1} + \zeta_{i2}$ where ζ'_{i1} is the sampled matched filtered and square law detector output of a first step resolution cell 'covering' the ith resolution cell of the second step and ζ_{i2} is the second step sampled matched filtered and square law detector output of the ith resolution cell. These decision statistics $\{ Z_i \}$ are as in the other adaptive mode now compared with thresholds

$$\left\{ T_i^{(J)} = K_J \hat{\sigma}_i^{\,2} \right\}$$

to decide upon the presence of a target. The estimate $\hat{\sigma}_i^{\,2}$ in this case is based on the use of second step resolution cell outputs surrounding the cell under test. So it is noted that the same false alarm probability regulation mechanism is employed for both modes. A schematic of this resolution-variant mode illustrating the general case where the number of resolution cells covering the admissible target parameter space on the second step is greater than the number of the first step because of the increased resolution employed is presented in Figure 4.

DETECTION PROBABILITY

The detection probability at a maximum range cell for both of these adaptive modes may be symbollically expressed as:

$$P_D = \sum_{J=1}^{M} P\left\{ \text{T.C.} \mid J, \bar{X} \right\} \quad P\left\{ J; \underline{\phi_0} \right\}$$

where the conditional probability of a threshold crossing event (T.C.) on the hypothesis that the Jth radar policy is employed, and $\bar{X}$ is the average received signal energy to clutter plus thermal noise power density, is expressed as:

$$P\left\{ \text{T.C.} \mid J, X \right\} = \int_{\beta_J = 0}^{\infty} P\left\{ z > \beta_J \mid J, X \right\} f(\beta_J \mid J)\, d\beta_J$$

where the change of variable $z = \frac{Z}{\sigma_0}$ and $\beta_J = \frac{T_J}{\sigma_0}$ is introduced.

$\sigma_0^{\,2}$ is the variance of the cell under test.

The conditional probability density function of the normalized threshold β_J on the hypothesis of the Jth policy is expressed as:

$$f(\beta_J \mid J)\, d\beta_J = \frac{1}{(R_J-1)!}\left(\frac{\beta_J}{C_J}\right)^{R_J-1} e^{-\beta_J/C_J}\ \frac{d\beta_J}{C_J},\quad C_J = \frac{K_J}{R_J}$$

and R_J represents the number of variates employed in the variance estimate. For the non-coherent gain second step $R_J = N_c\,(J - N_1)$, and for the resolution variant mode $R_J = N_c 2^{J-1}$.

For both the variable non-coherent gain mode and the resolution-variant mode, the expression of the detection probability is developed for the detection of a Swerling case #2 target model. For this target model, the target cross-sections are assumed to be statistically independent from one transmission to another. For the detection mode using a variable non-coherent integration gain on the second step, the decision statistic is

$$z = \sum_{i=1}^{J} \zeta_i$$

where

$$f(\zeta_i)\,d\zeta_i = \frac{1}{1+\overline{X}} \exp\left(\frac{-\zeta_i}{1+\overline{X}}\right) d\zeta_i \,.$$

$P\left\{ z > B_J \mid J, \overline{X} \right\}$ is shown to be:

$$P\left\{ z > \beta_J \mid J, \bar{X} \right\} = \exp\left(-\alpha\,\beta_J\right) \sum_{K=0}^{J-1} \frac{\alpha\,\beta_J}{K!}$$

where $\alpha = \dfrac{1}{1+\overline{X}}$, and finally

$$P\left\{ T.C \mid J, \bar{X} \right\} = \sum_{K=1}^{J-1} \frac{\left(\frac{\alpha\,(2K_J)}{R_J}\right)^K \left(R_J + K - 1\right)!}{\left(1 + \frac{\alpha(2K_J)}{R_J}\right)^{R_J + K} \left(K!\,(M-1)!\right)}$$

and $R_J = N_c\,(J - N_c)$.

For the resolution-variant detection mode, the decision statistic

$z = \zeta'_1 + \zeta_2$ where in this case $f_1(\zeta_1)\,d\zeta_1 = \dfrac{1}{1+X} \exp\left(\dfrac{\zeta_1}{1+\bar{X}}\right) d\zeta_1$

and $$f_2(\zeta_2)\,d\zeta_2 = \frac{1}{1+\overline{X}(2^{J-1})} \exp\left(\left(-\frac{\zeta_2}{1+\bar{X}\,2^{J-1}}\right)\right) d\zeta_2$$

The ratio of the average received signal energy to total noise power density for the received variate ζ_2 on the second sequential step is increased by the factor 2^{J-1}. The second step resolution is increased by this factor, and consequently, the effective output clutter power density of the second step is reduced by this same factor.

To obtain $f(z)\,dz$ for this resolution-variant case, the convolution of the density functions f_1 and f_2 are determined yielding the result:

$$f(z)\,dz = \frac{\exp\left(-\frac{z}{\lambda_J\bar{X}+1}\right) - \exp\left(\frac{-z}{\overline{X}+1}\right)}{(\lambda_J - 1)\,\overline{X}}\,dz \quad \text{where} \quad \lambda_J = 2^{J-1}.$$

$P\left\{ z > \beta_J \mid J, \overline{X} \right\}$ is then determined to be

$$P\left\{ z > \beta_J \mid J\ \overline{X} \right\} = \frac{\left(\lambda_J \overline{X} + 1\right) \exp\left(\frac{-\beta_J}{J\,x + 1}\right) - (\overline{X} + 1) \exp\left(\frac{-\beta_J}{\overline{x} + 1}\right)}{(\lambda_J - 1)\,\overline{X}}$$

and finally

$$P\left\{ T.C. \mid J, \overline{X} \right\} = \frac{(\lambda\,\overline{x} + 1)\left(\frac{\lambda_J\,\overline{X} + 1}{\lambda_J\,\overline{X} + C_J + 1}\right)^{R_J} - (\overline{X} + 1)\left(\frac{\overline{X} + 1}{\overline{X} + C_J + 1}\right)^{R_J}}{(\lambda_J - 1)\,\overline{X}}\,.$$

The development of the expression of the discrete probability of employing the Jth policy $P\{J; \underline{\phi_o}\}$ involves a relatively difficult problem of extreme value statistics with statistically dependent random variables. This is the case since the Jth policy is determined by the estimator $\hat{\phi}_{max} = \text{Max}\{\hat{\phi}_i\}$ where the $\hat{\phi}_i$ estimates depend on some common square law detector outputs. An approximation to the desired probability is obtained by assuming that the detection probability is computed for the cell containing the maximum clutter level estimate $\hat{\phi}_o$ and moreover that the estimate of this clutter level $\hat{\phi}_o$ is the maximum of the estimates $\{\hat{\phi}_i\}$.

The probability of employing the Jth policy $P\{J; \underline{\phi_o}\}$ may then be expressed as

$$P\{J; \underline{\phi_o}\} = P\left\{\phi_{T_J} < \hat{\phi}_o < \phi_{T_J}; \underline{\phi_o}\right\} \quad \text{or} \quad P\{J; \underline{\phi_o}\} = \int_{\phi_{T_{J-1}}}^{\phi_{T_J}} f(\hat{\phi}_o; \underline{\phi_o} d\hat{\phi}_o$$

The probability density function of the maximum likelihood estimate $\hat{\phi}_o$ is

$$f(\hat{\phi}_o)\, d\hat{\phi}_o = \frac{\left(\dfrac{N_c}{\underline{\phi_o}}\right)^{N_c} \hat{\phi}_o^{\,N_c - 1} \exp\left(-\dfrac{N_c}{\underline{\phi_o}} \hat{\phi}_o\right) d\hat{\phi}_o}{(N_c - 1)!}$$

where in this case the estimate of the true value $\underline{\phi_o}$ is made with a single transmission on the first sequential step, and N_c resolution cells are employed in the estimate

$$\hat{\phi}_o = \frac{2}{N_c N_o} \sum_{i=1}^{N_c} \frac{X_i^2}{W_i}$$

The discrete probability $P\{J; \underline{\phi_o}\}$ may then be expressed as:

$$P\{J; \underline{\phi_o}\} = F_G\left(\frac{N_c \phi_{T_J}}{\underline{\phi_o}}\right) - F_G\left(\frac{N_c \phi_{T_{J-1}}}{\underline{\phi_o}}\right)$$

$$\text{and } F_G(X) = \frac{1}{(N_c - 1)!} \int_0^X Y^{N_c - 1} \exp(-Y)\, dY .$$

FALSE ALARM PROBABILITY

The single cell false alarm probability at each of the second step resolution cells is expressed as

$$P_{FA} = \sum_{J=1}^{M} P\{T.C. \mid J, \overline{X} = 0\}\, P\{J; \underline{\phi_o}\} .$$

The conditional single cell false alarm probability $P\left\{ T.C. \mid J, \overline{X} = 0 \right\}$ on the hypothesis of the Jth resolution policy for the variable non-coherent integration gain mode and the resolution-variant mode are obtained from the equations of $P\left\{ T.C. \mid J, \overline{X} \right\}$ by setting $\overline{X} = 0$, and the expressions are:

For the variable non-coherent integration gain mode:

$$P\left\{ T.C. \mid J, \overline{X} = 0 \right\} = \sum_{K=0}^{J-1} \frac{(2\,C_J)^K \quad (R_J + K - 1)!}{(1 + 2\,C_J)^{R_J + K} \quad K! \quad (C_J - 1)!}$$

where for this case $R_J = N_C(J - N_1)$ and $C_J = \dfrac{K_J}{R_J}$ and for the resolution-variant mode:

$$P\left\{ T.C. \mid J, \overline{X} = 0 \right\} = \left(\frac{1}{C_J + 1} \right)^{R_J} \left(1 + \frac{R_J \; C_J}{1 + C_J} \right)$$

with $R_J = N_C \; 2^{J-1}$ and $C_J = \dfrac{K_J}{R_J}$.

It is noted that both of these single cell false alarm probabilities are not functions of the encountered clutter levels.

The set of threshold control proportionality constants $\left\{ K_J \right\}$ for the variable non-coherent integration gain mode are determined by a solution of the equation

$P\left\{ T.C. \mid J; \overline{X} = 0 \right\} = P_{F_0}$ for $J = 1, 2, --- M$ where P_{F_0} is the specified single cell false alarm probability. The single cell false alarm probability is then $P_{FA} = P_{F_0}$ since

$$\sum_{J=1}^{M} P\left\{ J; \phi_o \right\} = 1.$$

For the resolution-variant mode, the number of second-step resolution cells N_{F_2} covering the admissible target parameter space is a random variable since the cell size is dependent on the signal time duration-bandwidth product. ($N_{F_2} = N_{F_1} \; 2^{J-1}$ where N_{F_1} is the number of first sequential step resolution cells in the parameter space). Consequently, for this case the selection of the threshold proportionality constants $\left\{ K_J \right\}$ for the maintenance of a constant false alarm probability must be based on the total false alarm probability P_{FT} which is approximated as:

$$P_{FT} = \sum_{J=1}^{M} \left[1 - \left(1 - P\left\{ T.C. \mid J, \overline{X} = 0 \right\} \right)^{N_{F1} \; 2^{J-1}} \right] P\left\{ J; \underline{\phi_o} \right\}$$

This approximation is based on assuming that the event of a false alarm is statistically independent from one resolution cell to another. Actually the false alarm events are statistically dependent since the detection thresholds of resolution cells in close proximity to one another are controlled by some of the same observed random variables - the sampled square law detector outputs of the surrounding resolution cells which are employed in the variance estimates used in the threshold control. For the cases of interest where $P\{T.C. \mid J, \overline{X} = 0\} \ll 1$, the further approximation made is that:

$$P_{FT} \approx \sum_{J=1}^{M} N_F \; 2^{J-1} \; P\{T.C. \mid J, \overline{X} = 0\} \; P\{J, \phi_o\}$$

To keep the probability $P_{FT}' = \dfrac{P_{FT}}{N_F}$ equal to the specified value P_{FO} the threshold proportionality constant K_J is determined for $J = 1, 2, ---M$ by solving the equation

$$\frac{P\{T.C. \mid J, \overline{X} = 0\}}{2^{-J+1}} = P_{FO}' \quad \text{for } K_J \; .$$

The discrimination policy making thresholds $\{\phi_{T_J}\}$ are now set so that the detection probability P_D is maintained at a value greater than the specified lower bound P_{D_O}. Roughly speaking, the approach is to offset the fact that the conditional probability of detection $P\{T.C. \mid J, \overline{X}\}$ is a monotonically decreasing function of the clutter level $\underline{\phi_o}$ by making the degree of clutter discrimination J, a monotonically increasing function of $\underline{\phi_o}$ so that the lower bound P_{D_O} is maintained. This is accomplished for the computed cases by solving $P\{T.C. \mid J, \overline{X}\} = P_{D_O}$ for $\overline{X}_J$ for all $J = 1, 2, --- M\text{-}1$ and then solving for $\phi_{T_J} = \dfrac{\overline{X}_o}{\overline{X}_J}$. ($\overline{X}_o$ is the average received signal energy to thermal noise power density.) The first threshold ϕ_{T_1} has been set equal to unity.

Typical sets of computed detection probabilities and system costs based on the developed probability distributions are presented in Figures 5 and 6 for both adaptive detection modes as a function of $\underline{\phi_o}$ the ratio of the clutter plus thermal noise power density to the thermal noise power density.

ACKNOWLEDGMENT

The author gratefully acknowledges the help of Dr. R. S. Johnson.

REFERENCES

1. Westerfield, E.C., Proeger, R.H., and Stewart, J.L., "Processing Gain against Reverberation (Clutter) Using Matched Filters", I.R.E. Transactions on Inf. Theory Vol. II-6, June 1960.

2. Finn, H.M., "Adaptive Detection in Clutter", Proceedings of the National Electronics Conference, Volume XXII, 1966.

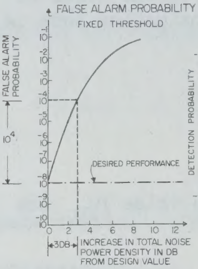


Figure 1A

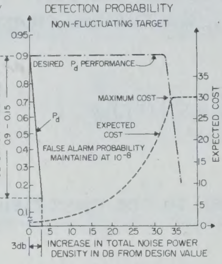


Figure 1B

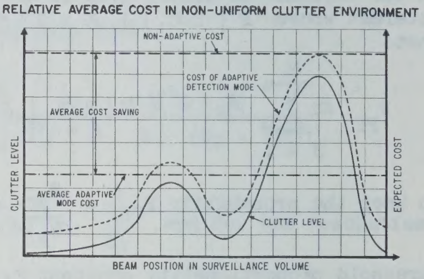


Figure 2

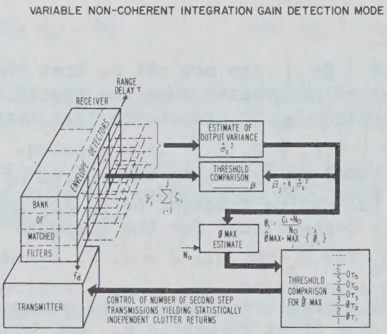

VARIABLE NON-COHERENT INTEGRATION GAIN DETECTION MODE
RANGE DELAY τ
RECEIVER
ENVELOPE DETECTORS
BANK OF MATCHED FILTERS
TRANSMITTER
fd
ESTIMATE OF OUTPUT VARIANCE
THRESHOLD COMPARISON
β
Ø MAX ESTIMATE
N₀
THRESHOLD COMPARISON FOR Ø MAX
CONTROL OF NUMBER OF SECOND STEP TRANSMISSIONS YIELDING STATISTICALLY INDEPENDENT CLUTTER RETURNS

Figure 3

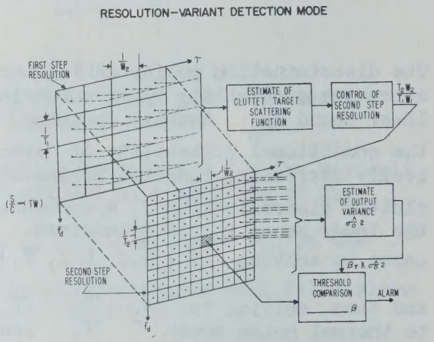

RESOLUTION-VARIANT DETECTION MODE
FIRST STEP RESOLUTION
ESTIMATE OF CLUTTET TARGET SCATTERING FUNCTION
CONTROL OF SECOND STEP RESOLUTION
ESTIMATE OF OUTPUT VARIANCE
SECOND STEP RESOLUTION
THRESHOLD COMPARISON
β
ALARM
fd

Figure 4

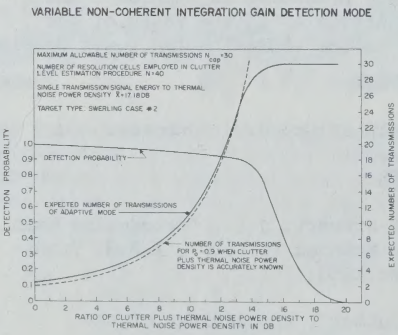


Figure 5

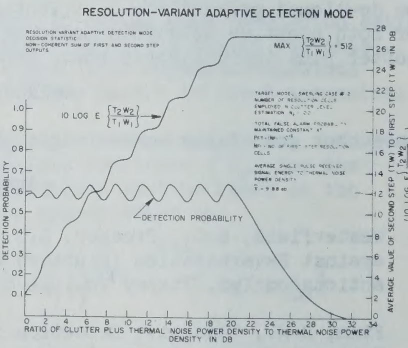


Figure 6

COMPUTER APPLICATIONS: CIRCUIT ANALYSIS

DONNAMAIE E. MEYERHOFF
Member of the Technical Staff, TRW Systems
Redondo Beach, California
Graduate Student
University of California at Los Angeles

ABSTRACT

The article gives a brief evaluation of several programs available to the circuit analyst, and outlines drawbacks common to them. In particular, a simple problem has been selected as a basis of comparison of the capabilities of some of the major programs: NASAP, ECAP, NET-1, SCEPTRE, PREDICT, and the TRW On-Line Computer System.

INTRODUCTION

Computers aid today's circuit designer by performing high-speed analysis of circuits through the application of general-purpose analysis programs. Computers evaluate changes effected by varying circuit parameters; simulate component failure; and optimize circuits, their layouts and interconnections. (1) Actual automated design, however, has not yet achieved a high degree of sophistication.

Some of the several hundred special and general-purpose programs dealing with circuit analysis will be discussed here. An attempt has been made to highlight the major programs and their capabilities. The programs have been reviewed primarily to evaluate their capability in the transient mode of analysis, since nonlinear transient analysis remains the most difficult problem the engineer-analyst encounters. (2)

The important considerations needed to select a program are computational accuracy; problem size restrictions; computational speed (and therefore, economics); problem flexibility; output format selection and interpretability; turn-around delay; equipment availability; and most important, user orientation. (3) The ideal computer analysis solution would contain only the desired output data, obtained in an efficient, economical manner.

DIGITAL OFF-LINE

The first programs developed for circuit analysis were for small, inexpensive, and lightly used computer systems. Sitting at the console, the engineer could interact directly, albeit inefficiently, with the computer. As computer systems became more complex, departments other than engineering made heavy demands on their services. The cost of the installation demanded highly efficient use which led to batch processing of computer runs. Batch processing requires that the engineer interface with a programmer, a coordinator, and computer administrative staff. Under these conditions the engineer no longer has direct contact with the computer and turn-around delay becomes a problem. (4, 1)

Batch processing, or off-line operation of a digital computer, is the most prevalent approach to circuit-analysis problems. Digital off-line programs for circuit analysis may be divided into two general groups, a) small and/or specialized design-analysis programs, and b) large, general-purpose programs, as shown in Figure 1.

Specialized, small programs are virtually limited to use by their creator, and are restricted to a special task or circuit and in problem size. Most existing programs fall into this classification. Because of these disadvantages, there has been a trend toward favoring the large

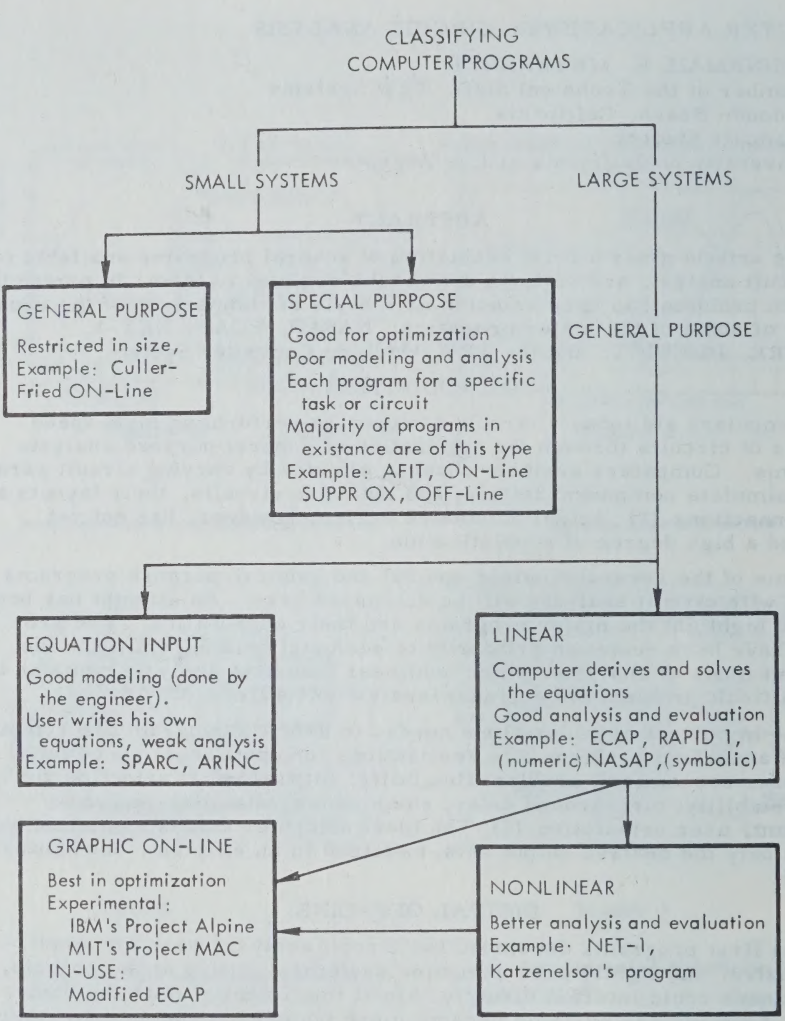

CLASSIFYING COMPUTER PROGRAMS
SMALL SYSTEMS
LARGE SYSTEMS
GENERAL PURPOSE
Restricted in size,
Example: Culler-Fried ON-Line
SPECIAL PURPOSE
Good for optimization
Poor modeling and analysis
Each program for a specific task or circuit
Majority of programs in existance are of this type
Example: AFIT, ON-Line
SUPPR OX, OFF-Line
GENERAL PURPOSE
EQUATION INPUT
Good modeling (done by the engineer).
User writes his own equations, weak analysis
Example: SPARC, ARINC
LINEAR
Computer derives and solves the equations
Good analysis and evaluation
Example: ECAP, RAPID 1,
(numeric) NASAP, (symbolic)
GRAPHIC ON-LINE
Best in optimization
Experimental:
IBM's Project Alpine
MIT's Project MAC
IN-USE:
Modified ECAP
NONLINEAR
Better analysis and evaluation
Example: NET-1,
Katzenelson's program

Figure 1.

Methods of generally classifying computer program development, based on an article by Arnold Spitalny, part two of the Electronics series. (Reference).

programs such as NET-1, ECAP, and SCEPTRE. The disadvantages of the general purpose programs include 1) restriction to analysis of circuit elements for which accurate models may be developed; 2) restriction in the size of the circuit to be analyzed (which limits the sophistication of the model); and 3) increasing program loading and execution times, and therefore, increasing costs of operation. (5, 3)

Program Comparison. There are two major categories in which circuit-analysis programs may be compared: 1) the program features and 2) the program capabilities. (3)

Capabilities. Capabilities of general-purpose programs should include linear and nonlinear dc analysis, ac analysis, linear and nonlinear transient analysis. These are Franklin Branin's groupings for numerical analysis techniques. Figure 2 shows the numerical methods presently available to solve problems in each of these categories. (1, 2, 6)

The first step in any computer-aided analysis is to derive a workable mathematical model to describe the system under study in the language of the computer. The accuracy of the analysis is limited by the accuracy of this model. (7) The computer models are generally formed from conventional circuit parameters. Most programs require the engineer to derive the device models (for example ECAP), while some use prestored models (for example NET-1).

Following the development of the equivalent circuit of the problem the circuit equations are derived. The favored programs automatically derive the equations. The methods they use are also indicated in Figure 2. The numerical methods of analysis are applied following the derivation of the circuit equations.

Features. According to Franklin Kuo (3), the features that a general network analysis program should include are:

1. User-oriented input
2. Modeling flexibility, including nonlinear capabilities
3. Flexible output format
4. Automatic parameter modifications
5. Error checks

These features are summarized in Figure 3 for the NET-1, ECAP, PREDICT, SCEPTRE, and NASAP programs.

Program Selection. After consideration of the capabilities and features of the various programs, one program must be selected for use with a given problem. Each program will have its own merits and each its own disadvantages. ECAP, NET-1, SCEPTRE, and PREDICT were selected as the leading examples of digital off-line programs since they meet a majority of the stated requirements for a general-purpose, circuit-analysis program.

ECAP was selected to represent the off-line classification of programs for the following reasons:

1. ECAP is considered more flexible than NET-1. (8, 9)
2. ECAP computation time is faster than PREDICT and SCEPTRE when the problem is small.
3. ECAP program loading time is faster than SCEPTRE and PREDICT.
4. ECAP accuracy is comparable to that of NET-1, PREDICT, and SCEPTRE, with the exception of junction devices (NET-1 specialty). (10)
5. ECAP is the most general program in analysis capability since it is the only one of the four capable of efficient ac analysis.

ECAP is not a perfect program. There has been observed deterioration in transient analysis routines when a problem exceeds 25 nodes. It is restricted to linear or piecewise linear simulation. Since output data generated by the program in the past has been excessive, new output

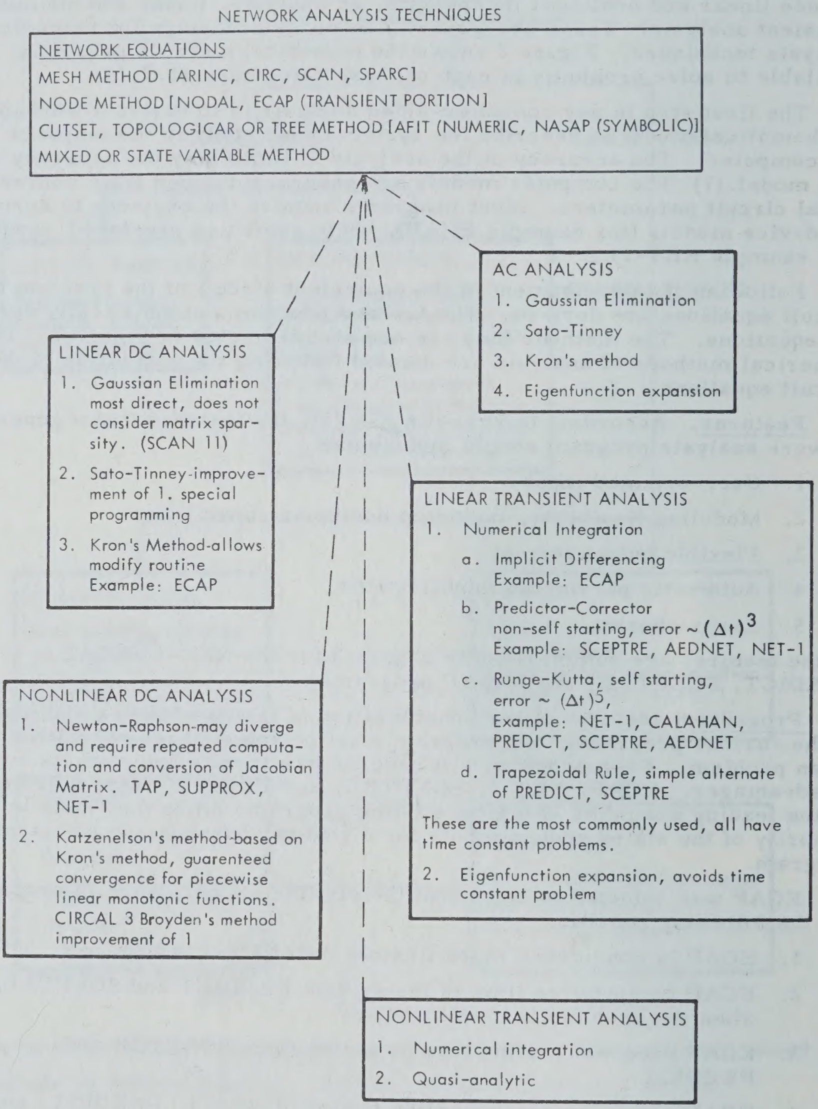

NETWORK ANALYSIS TECHNIQUES
NETWORK EQUATIONS
MESH METHOD [ARINC, CIRC, SCAN, SPARC]
NODE METHOD [NODAL, ECAP (TRANSIENT PORTION]
CUTSET, TOPOLOGICAR OR TREE METHOD [AFIT (NUMERIC, NASAP (SYMBOLIC)]
MIXED OR STATE VARIABLE METHOD
AC ANALYSIS
1. Gaussian Elimination
2. Sato-Tinney
3. Kron's method
4. Eigenfunction expansion
LINEAR DC ANALYSIS
1. Gaussian Elimination most direct, does not consider matrix sparsity. (SCAN 11)
2. Sato-Tinney-improvement of 1. special programming
3. Kron's Method-allows modify routine Example: ECAP
LINEAR TRANSIENT ANALYSIS
1. Numerical Integration
a. Implicit Differencing Example: ECAP
b. Predictor-Corrector non-self starting, error ~ $(\Delta t)^3$ Example: SCEPTRE, AEDNET, NET-1
c. Runge-Kutta, self starting, error ~ $(\Delta t)^5$, Example: NET-1, CALAHAN, PREDICT, SCEPTRE, AEDNET
d. Trapezoidal Rule, simple alternate of PREDICT, SCEPTRE
These are the most commonly used, all have time constant problems.
2. Eigenfunction expansion, avoids time constant problem
NONLINEAR DC ANALYSIS
1. Newton-Raphson-may diverge and require repeated computation and conversion of Jacobian Matrix. TAP, SUPPROX, NET-1
2. Katzenelson's method-based on Kron's method, guarenteed convergence for piecewise linear monotonic functions. CIRCAL 3 Broyden's method improvement of 1
NONLINEAR TRANSIENT ANALYSIS
1. Numerical integration
2. Quasi-analytic

Figure 2.

A summary of the various methods of solving the network equations. From Branin's article, (1, 2, and 6).

SUMMARY OF PROGRAM FEATURES

FEATURE	NET-1	ECAP	PREDICT	SCEPTRE	NASAP
User Orientated	Yes	Yes	Yes	Yes	No
Stored Models	Yes	No	No	Yes	No
Engineer-Derived Models	No	Yes	Yes	Yes	Yes
Piece Wise Linear Models	No	Yes	Yes	Yes	Yes
Nonlinear Subroutines	No	Yes	Yes	Yes	No
Add FORTRAN Statements	No	No	Yes	Yes	No
Automatic Parameter Modification	No	ac, dc	No	Rerun Feature	?
Sensitivity Analysis	No	Yes	No	No	Excellent
Error Checks	No	Node Imbalance	Same	Same	?
AC Analysis	No	Yes	No	No	Excellent
Transient Analysis	Matrix	Matrix	Matrix	Matrix	Dichotomy
Computer	MANIAC 7090/94	1620 7040/44 7090/94	7094 (IBM)	7094	CDC 3600 7094 Univac 1107
Language	FAP, MAP	FORTRAN II, IV	FORTRAN II	FORTRAN IV	FORTRAN IV

Figure 3.

A summary of the features available on five of the representative general purpose programs. Kuo (3) has stated that these are essential if the program is to be general purpose. References: (3, 1), program/manuals, also, "Charting the Program Jungle," editorial, EEE, Vol. XV, No. 2 (1967).

routines have been developed which allow tabulated and selective printing. Unlike SCEPTRE, however, the user may not add FORTRAN statements. This precludes directly usable output data in many cases.

Similarly to SCEPTRE and NET-1, there is no modify routine for the transient mode of analysis. Each value combination for a parameter-variation problem must be a separate, complete input card set with its own command and control cards.

Time-step selection is another difficulty. Δt must be selected by the engineer. Complicating the selection is the fact that a small Δt improves truncation errors; it also increases accumulated roundoff errors and computation time. SCEPTRE attempts to improve this problem by employing a variable time-step selection routine, with the engineer specifying the limits.

Common to all off-line programs are two difficulties: 1) long turn-around time delay and 2) keypunch delay and error. Turn-around delay is usually handled by running the engineer's problem overnight. This delay can be increased, however, when a keypunch or data error is made. As use of the "company" computer increases, higher priority jobs are expected to increase the engineer's delay time above 48 hours.

DIGITAL ON-LINE

The return to on-line operation, or direct interaction between the engineer and his computer, is the result of both the difficulties incurred by time delays and the high cost of the computer installation. The present developments are centered around a graphic interface on-line system. Since visual display allows realization of the full potential of an on-line system, the graphic interface is preferred over keyboard typewriter input alone.

To assure efficiency, time sharing of the computer is being developed with the capacity goal of 100 plus simultaneous users. The time-shared, on-line user is expected to: 1) know his problem and the solution objectives; 2) be competent in mathematical analysis and solution techniques; and 3) be familiar with both the program and the console equipment he is to use. The developers of the on-line programs must, therefore, keep user-oriented input procedures in mind.

The advantages of a graphic on-line system are its simplicity of input and the ability of easy error detection. The disadvantages of the on-line systems are the present high cost and the limited number of graphically compatible programs. (4, 5)

Another consideration is the length of computation time. When large circuits with complex input functions are to be analyzed, the computation time is lengthy, the direct interaction of man and machine is reduced, and the on-line loses its attractiveness. Total computation time will depend on the engineer's technique, the program he selects, and the computer hardware.

The on-line system with graphic interface requires a great deal of equipment (some of which is only now being developed) for adequate display and simplified input procedures. The output of the on-line system should be capable of graphic display as well as hard-copy printout, such as a CRT display coupled with a Calcomp plotter.

Large Systems. Large on-line systems are well represented by the AEDNET program operating on the project MAC time-sharing computer with the electronic systems display unit. The system is presently under research at MIT and is limited to 20 element circuits. (11, 12)

The system, which uses graphic input, allows the user to draw his circuit on the screen using a light pen. Values of the elements are input either by use of a keyboard or by "drawing" on the screen. The output is the waveform of various network variables. Modification of parameters or topology is possible at the end of each run.

The program accepts tabular, constant, or subroutine input for parameters, as does SCEPTRE (off-line). Modeling features are also comparable to SCEPTRE.

Small Systems. The small systems are well represented by the Culler-Fried On-line unit (specifically, TRW's system built by Bunker Ramo). The OLC (On-line Computer) is intended for use with problems requiring a significant amount of man-machine interaction. Its main advantage is the elimination of the long turn-around time delay. It is not intended to be a replacement for working programs already largely implemented, as is AEDNET. (13, 14, 15)

The OLC has four remote consoles, each equipped with two keyboards and a CRT display. The unit is a time-shared system, each user restricted to one-fourth of the memory. Other input/output facilities are a Calcomp plotter, an output typewriter (typing portion), and two magnetic tape units. A user may store symbolic subroutines and data, for recall at a future time, by use of an assigned code.

EXAMPLE PROBLEM

A simple problem, involving both parameter variation and restricted output data, was selected to demonstrate ECAP and the OLC. The four-parameter problem required a parameter variation through 390 combinations, and the desired output was a family of curves relating the analog conversion error to R_S, C_S at selected stabilization times. (The problem is a simplified multiplexer-source model.)

The problem and the input to the two systems is shown in Figure 4.

System Difficulties. ECAP required one complete card set for each of the 390 parameter variations. The OLC, following the engineer's derivation of a symbolic program, completed 127 parameter variations per computation run. ECAP required a great deal of keypunching, and detailed analysis for Δt selection. The OLC required a Δt selection also, as well as 1 hour of programming (button punching). ECAP required 2 hours of 7094 time at $700/hr while the OLC required 20 minutes of the smaller unit's time at $50/hr. ECAP output is excessive and not directly usable. The OLC output is directly usable and only the desired data is obtained.

CONCLUSIONS

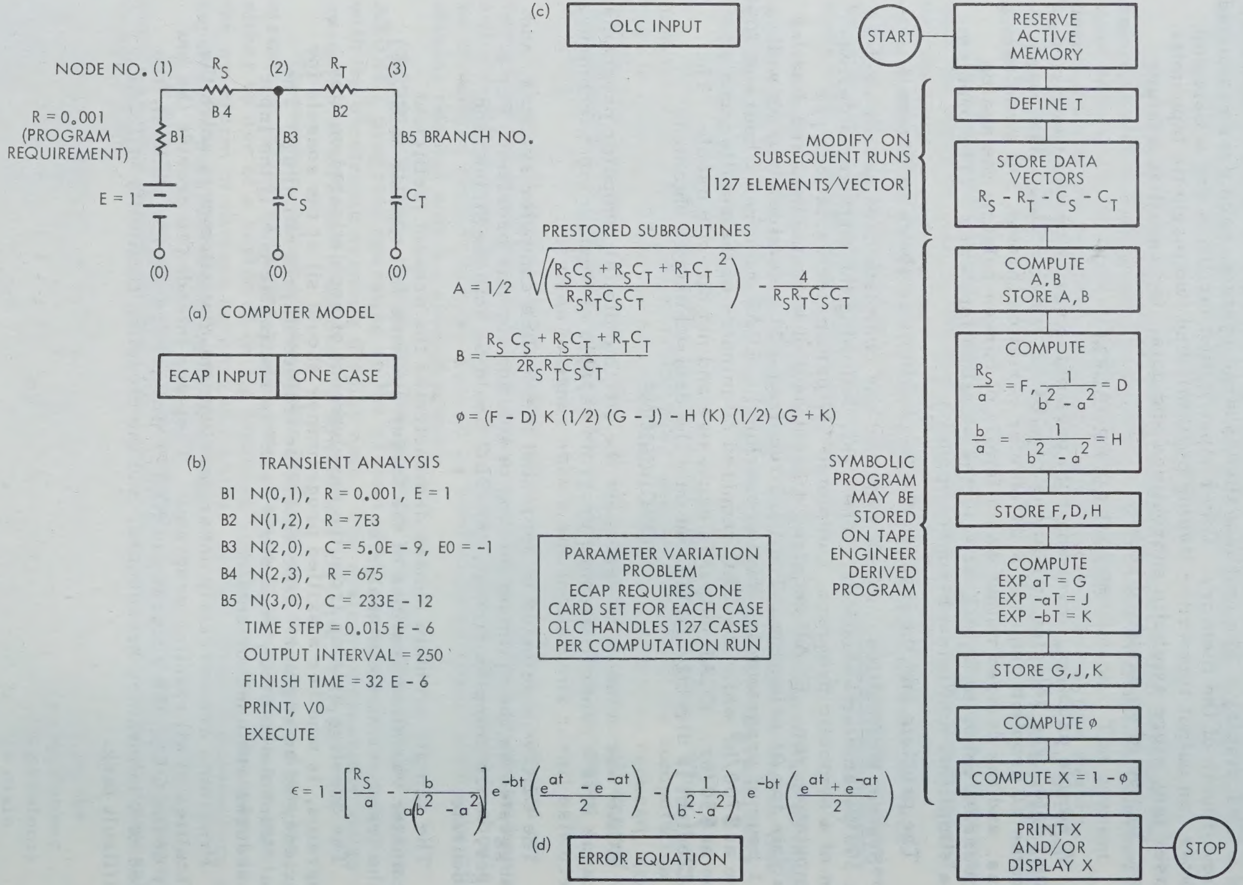
ECAP was selected to represent the off-line digital computer programs because it is a general-purpose program capable of flexible input techniques (piecewise linear simulation) and a wide range of analysis.

The OLC was selected to represent the on-line computer systems, and is suggested as the optimum method of solution for this problem. For a larger, more complex circuit, the OLC would be both inefficient and inaccurate.

The sample problem chosen demonstrates the present inefficient parameter variation routines of the larger systems (all similar to ECAP) in the transient mode of analysis.

The question arising from the development of sophisticated on-line programs, is whether a skilled programmer should sit at the console for efficiency or an engineer-user should be trained in programming. The ideal "console on every engineer's desk" is realizable only if the input procedures are user-orientated.

Programs are currently under development that attempt to solve the difficulties of all existing programs. They are intended to provide the low cost of the OLC, the accuracy of NET-1 (in junction-device studies), the speed of a laboratory breadboard, and the modeling flexibility of ECAP, a difficult task.


(a) COMPUTER MODEL
NODE NO. (1)
R_S
(2)
R_T
(3)
B4
B2
R = 0.001 (PROGRAM REQUIREMENT)
B1
B3
B5 BRANCH NO.
E = 1
C_S
C_T
(0)
(0)
(0)
ECAP INPUT
ONE CASE
(c)
OLC INPUT
START
RESERVE ACTIVE MEMORY
DEFINE T
MODIFY ON SUBSEQUENT RUNS
127 ELEMENTS/VECTOR
STORE DATA VECTORS $R_S - R_T - C_S - C_T$
PRESTORED SUBROUTINES
COMPUTE A, B STORE A, B
COMPUTE
SYMBOLIC PROGRAM MAY BE STORED ON TAPE ENGINEER DERIVED PROGRAM
STORE F, D, H
COMPUTE EXP aT = G EXP -aT = J EXP -bT = K
STORE G, J, K
COMPUTE φ
COMPUTE X = 1 - φ
PRINT X AND/OR DISPLAY X
STOP
(d)
ERROR EQUATION

(b) TRANSIENT ANALYSIS

```
B1  N(0,1),  R = 0.001, E = 1
B2  N(1,2),  R = 7E3
B3  N(2,0),  C = 5.0E - 9, E0 = -1
B4  N(2,3),  R = 675
B5  N(3,0),  C = 233E - 12
    TIME STEP = 0.015 E - 6
    OUTPUT INTERVAL = 250
    FINISH TIME = 32 E - 6
    PRINT, V0
    EXECUTE
```

$$A = 1/2 \sqrt{\left(\frac{R_S C_S + R_S C_T + R_T C_T}{R_S R_T C_S C_T}\right)^2 - \frac{4}{R_S R_T C_S C_T}}$$

$$B = \frac{R_S C_S + R_S C_T + R_T C_T}{2 R_S R_T C_S C_T}$$

$$\phi = (F - D)\, K\, (1/2)\, (G - J) - H\, (K)\, (1/2)\, (G + K)$$

$$\frac{R_S}{a} = F, \quad \frac{1}{b^2 - a^2} = D$$

$$\frac{b}{a} = \frac{1}{b^2 - a^2} = H$$

PARAMETER VARIATION PROBLEM
ECAP REQUIRES ONE CARD SET FOR EACH CASE
OLC HANDLES 127 CASES PER COMPUTATION RUN

$$\epsilon = 1 - \left[\frac{R_S}{a} - \frac{b}{a\left(b^2 - a^2\right)}\right] e^{-bt} \left(\frac{e^{at} - e^{-at}}{2}\right) - \left(\frac{1}{b^2 - a^2}\right) e^{-bt} \left(\frac{e^{at} + e^{-at}}{2}\right)$$

Figure 4.
Input Data for Parameter Variation.

BIBLIOGRAPHY

1. Donald Christiansen, "The Man-Machine Merger," Electronics, pp. 110-123, September 19, 1966.
2. Franklin H. Branin, Jr., "Analyzing Circuits by the Numbers," Electronics, pp. 88-102, January 9, 1967.
3. Franklin F. Kuo, "Network Analysis by Digital Computer," Proc. IEEE, Vol. 54, pp. 820-829, June 1966.
4. Walter J. Karplus, ed., On-Line Computing, New York: McGraw-Hill, 1967.
5. Jim Kendler, "Computer-Aided Circuit Design," EEE, Vol. 15, pp. 92-97, June 1967.
6. Jacques J. Vidal and Walter J. Karplus, Lecture Notes: Computer Applications; Ordinary Differential Equations, (mimeo-course 124A), the University of California at Los Angeles, 1966.
7. Cyrus O. Harbourt, "Doing a Model Job," Electronics, pp. 82-87, January 23, 1967.
8. Donald Christiansen, "Comparing the 'Big Two' Programs," Electronics, pp. 74-75, February 6, 1967.
9. Herbert M. Wall, "Flexibility is ECAP's Forte," Electronics, pp. 82-89, February 6, 1967.
10. Allan F. Malmberg, "NET-1 Gets and 'A' for Accuracy," Electronics, pp. 76-82, February 6, 1967.
11. R. M. Fano, "The MAC System: The Computer Utility Approach," IEEE Spectrum, Vol. 2, pp. 56-64, January 1965.
12. J. Katzenelson, "AEDNET: A Simulator for Nonlinear Networks," Proc. IEEE, Vol. 54, pp. 1536-1552, November 1966.
13. Burton D. Fried, The TRW On-Line Computer, Vol. 1, California: TRW, 1964. (No. 9824-6001-RU-000 Physical Research Division.)
14. C. C. Farrington and D. Pope, The TRW On-Line Computer, Vol. 2, California: TRW, 1964. (No. 9824-6001-RU-000 Physical Research Division.)
15. Burton D. Fried, On-Line Problem Solving, California: TRW, 1966. (No. 9863-6001-R0000 Physical Research Center.)

ACKNOWLEDGEMENT

The author wishes to express her appreciation to Professor Walter J. Karplus and Professor Lawrence McNamee of the Engineering Department of the University of California at Los Angeles for their valuable suggestions.

REFERENCES

IBM 1620 Electronic Circuit Analysis Program (ECAP) Users' Manual (Note: now 7090/94), New York: IBM, 1965.

The Service Bureau Corporation 7090/7094 ECAP-Level No. 3, H 20-0170-1, 1966.

Richard Carpenter and William Happ, "Analyzing Circuits with Symbols," Electronics, pp. 92-98, December 12, 1966.

Allan F. Malmberg, et al., NET-1 Network Analysis Program 7090/94 Version, New Mexico: University of California at Los Alamos, 1964.

Harry W. Mathers, et al., Automated Digital Computer Program for Determining Responses of Electronic Circuits to Transient Nuclear Radiation (SCEPTRE), Vol. I and Vol. II, New York: IBM, February 1967.

Automated Digital Computer Program for Determining Responses of Electronic Systems to Transient Nuclear Radiation (PREDICT Circuit Analysis Program), New Mexico: Air Force Weapons Laboratory, Revised edition, September 1965.

George Rostky, ed., "Charting the Program Jungle," EEE, Vol. 15, pp. 28-35, May 1967.

Arnold Spitalney, "The Computer Excels as Analyst," Electronics, pp. 68-74, November 28, 1966.

APID COMPUTER TIME RESPONSE CALCULATION FOR SYSTEMS WITH ARBITRARY INPUT SIGNALS

CHRISTOPHER POTTLE
Cornell University
Ithaca, New York

ABSTRACT

In computing the response of linear systems which are ill-conditioned or are driven with signals known only as time series, the usual partial fraction expansion technique is not feasible. In such cases one usually resorts to numerical integration. A well-known numerical integration formula is proposed here and is compared with the more familiar Runge-Kutta and Trapezoidal formulas.

INTRODUCTION

The methods now available for computer calculation of the time response of lumped, linear, time-invariant systems are too numerous to mention here. Suffice it to say that each is to be recommended under certain conditions. Practical problems to which this paper apply are typically of the following sort: Consider a communication system in block diagram form, perhaps containing memoryless nonlinearities, which is being optimized with a time domain performance index using a digital computer. The adjustable parameters in the optimization procedure are the pole and zero positions of the linear portions of the system. Since all useful optimization techniques require repeated analysis of the system, a fast method of computing time response given sampled input signal values is required to simulate system behavior on the computer. Even if the entire system is linear and the excitation is representable as the impulse response of a linear time-invariant system, the problem may arise that, due to the degree of the system or the proximity of poles to one another, the partial fraction expansion is difficult to carry out with any precision. The usual approach to such problems is to resort to numerical integration techniques such as Runge-Kutta applied directly to the state equations of the system or the Trapezoidal Rule obtained by applying the bilinear Z-transformation to the system transfer function [1]. The former is accurate for fairly large integration intervals but is very slow; the latter is fast, but requires small integration steps to achieve any accuracy.

In this paper still another numerical method is presented - one which is well-known, but apparently has not been applied previously in this area. Its error term is proportional to the fourth power of the integration step size used, and is thus to be compared with Runge-Kutta methods. Its accuracy gives it an advantage over the Trapezoidal Rule whose error term is proportional to the square of the step size. It is assumed that any "best" method, particularly when an iterative optimization technique is involved, will be that which, for a prescribed accuracy or a prescribed input signal sampling interval, uses the least computer time. After a derivation of the numerical technique as applied to linear systems, the results of several computer experiments are presented to illustrate the superiority of the proposed method over those currently in use in such problems.

SYSTEM REPRESENTATION

It will be assumed here that the problem at hand is to obtain the time response of a single-input single-output linear time-invariant system whose system function is given in terms of its poles and zeros. The input signal either is representable as the impulse response of another such system or is in sampled-data form. The system function, then, can be written in the form

$$H(s) = G \frac{\prod_{i=1}^{m} (s - z_i)}{\prod_{j=1}^{n} (s - p_j)} \quad . \tag{1}$$

It will be assumed further that $m \leq n$. If $m = n$, we will work with an $H_1(s)$ where

$$H_1(s) = H(s) - G \tag{2}$$

whose zeros are assumed known. With $H(s)$ in this form a cascade representation of the system is possible, as shown in block form in Fig. 1 .

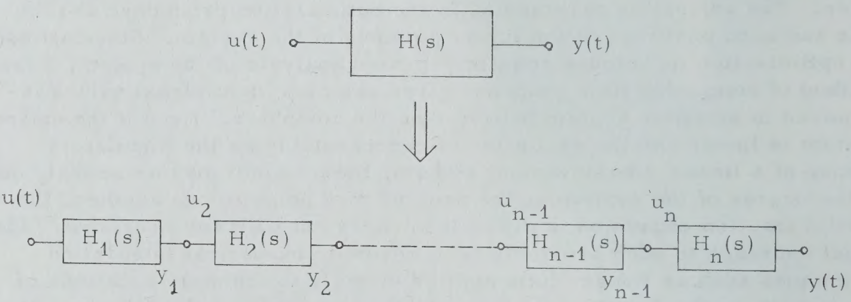


Fig. 1. Block Diagram of Cascade System Representation.

Each of the subsystems in the cascade is of first or second order for reasons to be discussed later, so that the k^{th} link is a system whose transfer function can be written in one of the five following forms:

$$G_k \frac{(s - z_1^{(k)})(s - z_2^{(k)})}{(s - p_1^{(k)})(s - p_2^{(k)})} \qquad G_k \frac{s - z_1^{(k)}}{(s - p_1^{(k)})(s - p_2^{(k)})}$$

$$\frac{G_k}{(s - p_1^{(k)})(s - p_2^{(k)})} \qquad G_k \frac{s - z_1^{(k)}}{s - p_1^{(k)}} \qquad \frac{G_k}{s - p_1^{(k)}} \quad . \tag{3}$$

In quotient-of-polynomial form each subsystem has the general representation

$$H_k(s) = \frac{b_2 s^2 + b_1 s + b_0}{s^2 + a_1 s + a_0} \quad . \tag{4}$$

In the sequel, the subscript k will be dropped except where needed for clarity. The differential equation relating the input and output of a system with this transfer function is

$$\ddot{y}(t) + a_1 \dot{y}(t) + a_0 y(t) = b_2 \ddot{u}(t) + b_1 \dot{u}(t) + b_0 u(t) \quad . \tag{5}$$

With the introduction of a suitable two-element state vector x, the state and output-state equations of this system may be obtained in so-called phase-variable form:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \tag{6}$$

$$y(t) = \begin{bmatrix} b_0 - a_0 b_2 & b_1 - a_1 b_2 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + b_2 u(t) \quad . \tag{7}$$

These equations may be verified by differentiating (7) twice and summing to obtain the left-hand side of (5), using (6) to remove derivatives of x_1 and x_2. A minor calculation brings the output-state equation into the form

$$y(t) = \begin{bmatrix} b_0 & b_1 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + b_2 \dot{x}_2(t) \quad . \tag{8}$$

A block diagram of two adjacent subsystems realizing (6) and (8) as a collection of integrators, scalors, and summers is given in Fig. 2 .

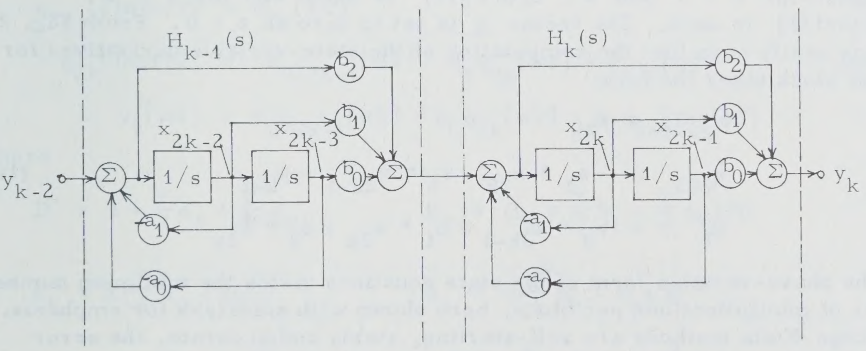


Fig. 2. Block Diagram of Two Adjacent Subsystems.

TWO FAMILIAR COMPUTATIONAL ALGORITHMS

A useful way of comparing computational methods for obtaining the time response of continuous systems in stepwise fashion is to observe that they all convert the original differential equations describing the system into difference equations. The solution of these difference equations represents the response of a discrete system which approximates the behavior of the original system at a set of points evenly spaced in time. In this section the discrete approximations obtained by the application of two familiar numerical algorithms will be stated and their properties discussed.

Runge-Kutta Methods

The name Runge-Kutta is attached to a class of methods of which perhaps the most popular is due to Gill [2]. One step of this algorithm consists of carrying out the following calculations four times, for $i = 0(1)3$:

$$\begin{aligned} \underline{z}^{(i)} &= T * \underline{\dot{x}}^{(i)} \\ \underline{r}^{(i)} &= a^{(i)} * (\underline{z}^{(i)} - b^{(i)} * \underline{q}^{(i)}) \\ \underline{x}^{(i+1)} &= \underline{x}^{(i)} + \underline{r}^{(i)} \\ \underline{q}^{(i+1)} &= \underline{q}^{(i)} + 3 * \underline{r}^{(i)} + c^{(i)} * \underline{z}^{(i)} \end{aligned} \qquad (9)$$

where

$$\begin{array}{lll} a_0 = 1/2 & b_1 = 2 & c_1 = -1/2 \\ a_1 = 1 - \sqrt{2}/2 & b_2 = 1 & c_2 = -1 + \sqrt{2}/2 \\ a_2 = 1 + \sqrt{2}/2 & b_3 = 1 & c_3 = -1 - \sqrt{2}/2 \\ a_3 = 1/6 & b_4 = 2 & c_4 = -1/2 \end{array}$$

If $\underline{x}^{(0)}$ in the formula above is $\underline{x}[nT]$, then $\underline{x}^{(3)}$ is $\underline{x}[(n+1)T]$, where the derivative vector $\underline{\dot{x}}^{(i)}$ is that obtained from the state equation (6) using $\underline{x}^{(i)}$ as the state vector. For $i = 0$, $u[nT]$ is used as the input signal; for $i = 1$ and 2, $u[(n+\frac{1}{2})T]$ is used; and for $i = 3$, $u[(n+1)T]$ is used. The vector $\underline{q}$ is set to zero at $t = 0$. From Fig. 2 it is easily seen that the computation of the state-variable derivatives for one block takes the form

$$\begin{aligned} \dot{x}_{2k-1} &= x_{2k} \\ \dot{x}_{2k} &= -a_1 * x_{2k} - a_0 * x_{2k-1} + y_{k-1} \\ y_k &= b_0 * x_{2k-1} + b_1 * x_{2k} + b_2 * \dot{x}_{2k} \end{aligned} \quad . \qquad (10)$$

The phase-variable form of the state equations yields the minimum number (5) of multiplications per block, here shown with asterisks for emphasis. Runge-Kutta methods are self-starting, stable and accurate, the error term being proportional to T^4. Their great disadvantage is the number of arithmetic operations necessary to complete one time step (here 60 multiplications and 48 additions per block).

The Trapezoidal Rule

The Trapezoidal Rule refers to the formula for the approximation of a definite integral given below:

$$\int_{nT}^{(n+1)T} g(t)\,dt \approx \frac{T}{2}\left(g[(n+1)T] + g[nT]\right) \quad . \tag{11}$$

In this case the difference equation of the approximate discrete subsystem is most easily and quickly obtained from the discrete transfer function resulting when the bilinear z - form

$$\frac{2}{T}\;\frac{z-1}{z+1}$$

is substituted for s in the original transfer function [1] . A more direct method is to integrate the state equation itself using the trapezoidal rule:

$$\int_{nT}^{(n+1)T} \dot{\underline{x}}(t)\,dt = \int_{nT}^{(n+1)T} [A\underline{x}(t) + Bu(t)]\,dt \tag{12}$$

or

$$\underline{x}[(n+1)T] - \underline{x}[nT] = \frac{T}{2}\{A\,\underline{x}[(n+1)T] + A\,\underline{x}[nT]\}$$

$$+ \frac{T}{2}\{Bu[(n+1)T] + Bu[nT]\} \quad . \tag{13}$$

This equation when solved for $\underline{x}\,[(n+1)T]$ yields the difference equation

$$\underline{x}[(n+1)T] = [I - \frac{T}{2}A]^{-1}\{[I + \frac{T}{2}A]\,\underline{x}[nT]$$

$$+ \frac{T}{2}(Bu[(n+1)T] + Bu[nT])\} \quad . \tag{14}$$

A fairly complicated transformation of the state vector yields the state equation of the k^{th} discrete subsystem in phase-variable form with transformed state vector $\underline{q}$.

$$\begin{aligned} q_{2k-1}[(n+1)T] &= q_{2k}[nT] \\ q_{2k}[(n+1)T] &= -\alpha_0 q_{2k-1}[nT] - \alpha_1 q_{2k}[nT] + y_{k-1}[nT] \\ y_k[nT] &= \beta_0 q_{2k-1}[nT] + \beta_1 q_{2k}[nT] + \beta_2 q_{2k}[(n+1)T] \end{aligned} \tag{15}$$

where

$$D = 1 + \frac{T}{2}a_1 + \frac{T^2}{4}a_0 \qquad \beta_2 = (b_2 + \frac{T}{2}b_1 + \frac{T^2}{4}b_0)/D$$

$$\alpha_1 = 2(\frac{T^2}{4}a_0 - 1)/D \qquad \beta_1 = 2(\frac{T^2}{4}b_0 - b_2)/D \quad .$$

$$\alpha_0 = (1 - \frac{T}{2}a_1 + \frac{T^2}{4}a_0)/D \qquad \beta_0 = (b_2 - \frac{T}{2}b_1 + \frac{T^2}{4}b_0)/D$$

The block diagram of this system is identical to that of Fig. 2 with $1/z$ for $1/s$, α_i for a_i, and β_i for b_i. The state vector transformation involves the input, with the result that the new initial state vector is not necessarily zero but rather

$$q_{2k-1}(0) = \frac{D}{4} y_{k-1}(0) \qquad q_{2k}(0) = -\frac{D}{4} y_{k-1}(0) \quad . \tag{16}$$

Similar relationships can be derived for a subsystem with only one pole. The simulation of the original differential system goes very rapidly using the trapezoidal rule in phase variable form, since only 5 multiplications and 4 additions are required per time step per subsystem. The error term of the approximation, however, is proportional to T^2, meaning that for high accuracy, very small steps may be necessary.

A PROPOSED TECHNIQUE

The trapezoidal approximation of a definite integral results when the integrand is interpolated at the upper and lower limits by a polynomial of degree one and the polynomial integrated. The proposed method assumes both the integrand and its derivative is known at the upper and lower limits. In this case a cubic can be interpolated and its integral found. The error term in this procedure is thus proportional to T^4. The resulting formula is

$$\int_{nT}^{(n+1)T} g(t)dt \approx \frac{T}{2}\left\{ g[nT] + g[(n+1)T] + \frac{T}{6}\left[\dot{g}[nT] - \dot{g}[(n+1)T]\right]\right\} \quad . \tag{17}$$

This formula is well known in numerical analysis [3]. Although a direct development could be carried out as in the last section, a different approach was taken. The solution in difference equation form of the state equation (6) of one subsystem is

$$\underline{x}[(n+1)T] = e^{AT}\underline{x}[nT] + \int_{nT}^{(n+1)T} e^{A[(n+1)T-\tau]} Bu(\tau)d\tau \quad . \tag{18}$$

For a system with a one- or two- element state vector the state transition matrix e^{AT} is easily obtained exactly for all possible cases. The convolution integral, however, must in general be approximated and it is to this definite integral that the formula (17) is applied. In this case,

$$\begin{aligned} g(\tau) &= e^{A[(n+1)T - \tau]} B\,u(\tau) \\ \dot{g}(\tau) &= e^{A[(n+1)T - \tau]} [B\,\dot{u}(\tau) - A\,B\,u(\tau)] \end{aligned} \quad . \tag{19}$$

The difference equation (18) then becomes (approximately)

$$\begin{aligned} \underline{x}[(n+1)T] = e^{AT}\left\{ \underline{x}[nT] + \frac{T}{2}\left[(I - \frac{T}{6}A)\,B\,u[nT] + \frac{T}{6}B\,\dot{u}[nT]\right]\right\} \\ + \frac{T}{2}\left[(I + \frac{T}{6}A)\,B\,u[(n+1)T] - \frac{T}{6}B\,\dot{u}[(n+1)T]\right] \quad . \end{aligned} \tag{20}$$

The output-state equation remains as before

$$y[nT] = C\,\underline{x}[nT] + D\,u[nT] \quad . \qquad (21)$$

The output derivative is now required as the input derivative to the next subsystem

$$\begin{aligned} \dot{y}[nT] &= C\,\dot{\underline{x}}[nT] + D\,\dot{u}[nT] \\ &= C\,A\,\underline{x}[nT] + C\,B\,u[nT] + D\,\dot{u}[nT] \quad . \qquad (22) \end{aligned}$$

A transformation of the state vector is now introduced to remove input derivatives and $u[(n+1)T]$. Let

$$\underline{q}[nT] = \frac{1}{T}\underline{x}[nT] - \frac{1}{2}\left[(I+\frac{T}{6}A)\,B\,u[nT] - \frac{T}{6}\,B\,\dot{u}[nT]\right] \quad . \qquad (23)$$

Then $\underline{q}[(n+1)T] = e^{AT}\{\underline{q}[nT] + B\,u[nT]\}$

$$\begin{aligned} y[nT] &= T\,C\,\underline{q}[nT] + \{\frac{T}{2}\,C(I+\frac{T}{6}A)\,B + D\}\,u[nT] \\ &\qquad - \frac{T^2}{12}\,C\,B\,\dot{u}[nT] \quad . \qquad (24) \\ \dot{y}[nT] &= T\,C\,A\,\underline{q}[nT] + \{\frac{T}{2}\,C\,A(I+\frac{T}{6}A)\,B + C\,B\}\,u[nT] \\ &\qquad - \{\frac{T^2}{12}\,C\,A\,B - D\}\,\dot{u}[nT] \end{aligned}$$

Introducing the special structure possessed by each subsystem,

$$A = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad C = [\,b_0 - a_0 b_2 \quad b_1 - a_1 b_2\,] \quad D = b_2 \,,$$

and letting $e^{AT} = \begin{bmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{bmatrix}$

the state and output-state equations of the approximating discrete subsystems are obtained.

$$\begin{aligned} q_{2k-1}[(n+1)T] &= \phi_{11} * q_{2k-1}[nT] + \phi_{12} * \{q_{2k}[nT] + y_{k-1}[nT]\} \\ q_{2k}[(n+1)T] &= \phi_{21} * q_{2k-1}[nT] + \phi_{22} * \{q_{2k}[nT] + y_{k-1}[nT]\} \\ y_k[nT] &= \gamma_1 * q_{2k-1}[nT] + \gamma_2 * \{q_{2k}[nT] - \frac{T}{12}\dot{y}_{k-1}[nT]\} \\ &\qquad + \gamma_3 * y_{k-1}[nT] \qquad (25) \\ \frac{T}{12}\dot{y}[nT] &= \epsilon_1 * q_{2k-1}[nT] + \epsilon_2 * \{q_{2k}[nT] - \frac{T}{12}\dot{y}_{k-1}[nT]\} \\ &\qquad + \epsilon_3 * y_{k-1}[nT] + \epsilon_4 * \frac{T}{12}\dot{y}_{k-1}[nT] \end{aligned}$$

where

$$\gamma_1 = T(b_0 - a_0 b_2) \qquad \epsilon_1 = -\frac{T}{12} a_0 \gamma_2$$

$$\gamma_2 = T(b_1 - a_1 b_2) \qquad \epsilon_2 = \frac{T}{12}(\gamma_1 - a_1 \gamma_2) \; .$$

$$\gamma_3 = \frac{T}{12}\gamma_1 + \left(\frac{1}{2} - \frac{T}{12} a_1\right)\gamma_2 + b_2 \qquad \epsilon_3 = \frac{T}{12}\epsilon_1 + \left(\frac{1}{2} - \frac{T}{12} a_1\right)\epsilon_2 + \frac{1}{12}\gamma_2$$

$$\epsilon_4 = b_2$$

A block diagram of this second order system is given in Fig. 3 . The number of multiplications (shown by asterisks) required per time step per subsystem is 11 . A transformation to phase variable form was not tried here because it could save at the most one multiplication and no additions.

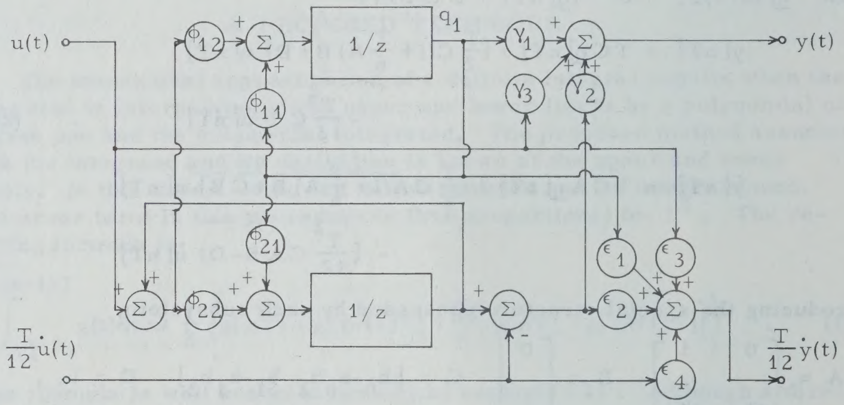


Fig. 3. Block Diagram of Discrete Approximation to Continuous Subsystem after Application of Derivative Method.

At a cost of about twice the number of operations, then, this method has an error term proportional to T^4 as compared to the trapezoidal rule's T^2 . It should be noted, however, that each of the discrete blocks has two inputs and two outputs, which is consistent with neither the single input signal nor the single output signal desired. The input signal problem is resolved by approximating the convolution integral of the first block by Simpson's Rule, another T^4 method. After the transformation

$$\underline{q}[nT] = \frac{3}{T}\underline{x}[nT] - \frac{1}{2} B\, u[nT] \tag{26}$$

the state and output-state equations of the first block appear as

$$\begin{aligned} q_1[(n+1)T] &= \phi_{11} q_1[nT] + \phi_{12}\,\{q_2[nT] + u[nT]\} + 2\,\hat{\phi}_{12}\, u[(n+\tfrac{1}{2})T] \\ q_2[(n+1)T] &= \phi_{21} q_1[nT] + \phi_{22}\,\{q_2[nT] + u[nT]\} + 2\,\hat{\phi}_{22}\, u[(n+\tfrac{1}{2})T] \\ y_1[nT] &= \gamma_1\, q_1[nT] + \gamma_2\,\{q_2[nT] + \tfrac{1}{2} u[nT]\} \\ \frac{T}{12}\dot{y}_1[nT] &= \epsilon_1\, q_1[nT] + \epsilon_2\,\{q_2[nT] + \tfrac{1}{2} u[nT]\} + \tfrac{1}{4}\gamma_2\, u[nT] \end{aligned} \tag{27}$$

where $e^{AT/2} = \begin{bmatrix} \hat{\phi}_{11} & \hat{\phi}_{12} \\ \hat{\phi}_{21} & \hat{\phi}_{22} \end{bmatrix}$

$$\gamma_1 = \frac{T}{3} b_0 \qquad \epsilon_1 = -\frac{T}{12} a_0 \gamma_2 \quad .$$

$$\gamma_2 = \frac{T}{3} b_1 \qquad \epsilon_2 = \frac{T}{12} (\gamma_1 - a_1 \gamma_2)$$

It is assumed that the first block represents a strictly proper system; i.e., the number of poles exceeds the number of zeros, and $b_2 = 0$. In this manner input signal derivatives are not required, but rather the signal sample in mid-interval, as in the Runge-Kutta procedure. This sample may be restored at the final system output by using a form of the proposed integration formula suited for interpolation rather than integration [3] :

$$y[(n+\frac{1}{2})T] \approx \frac{1}{2} \{y[nT] + y[(n+1)T] + \frac{T}{4} (\dot{y}[nT] - \dot{y}[(n+1)T])\} \quad . \qquad (28)$$

Since the state transformation involves the input signal, it is again possible for the initial state $\underline{q}(0)$ to be other than zero. Applying the transformations given earlier the initial state $\underline{q}(0)$ becomes

Block 1 Other Blocks

$$q_1(0) = 0 \qquad q_{2k-1}(0) = -\frac{T}{12} y_{k-1}(0)$$

$$q_2(0) = -\frac{1}{2} u(0) \qquad q_{2k}(0) = -(\frac{1}{2} - \frac{T}{12} a_1) y_{k-1}(0) + \frac{T}{12} \dot{y}_{k-1}(0) \quad . \qquad (29)$$

TEST RESULTS

The three numerical algorithms mentioned in this report were programmed in FORTRAN IV for both the PDP-7 and the 7094 computers. Every attempt was made within the FORTRAN framework to optimize these routines so a comparison would be meaningful. Identical results were not expected from the two computers, since their method of doing floating-point arithmetic is different. The 7094 carries out such operations directly in hardware at very high speed, so that bookkeeping and indexing occupy a significant part of the execution time. The PDP-7, however, does floating point arithmetic interpretively using 3 words of storage per variable. The result is that over 99% of the execution time is spent doing floating-point arithmetic. The example used here to compare these methods is a five-pole "Schussler filter", one of a class of filters designed to have a fast rise time in response to a step input together with small overshoot and restricted bandwidth [4] . The step response was normalized to one, and the maximum deviation from the correct response was used as an error measure. This error as a function of integration step size is shown in Fig. 4 . The difference between the second order and the two fourth order methods is immediately obvious, while the proposed method involving derivatives is seen to be only slightly more accurate than the Runge-Kutta method. Any difference between computers can only be explained by the different word lengths of the two machines. The significant advantage of the proposed method appears in Fig. 5 , where the actual relative running times of these algorithms are compared. These results indicate that the Trapezoidal Rule is to be recommended only when an error greater than 4% can be tolerated. Further experimentation suggests that the proposed method is (in general) superior to the others when errors of less than 1% are desired. There appears to be little difference in the results obtained on the two computers. When a sampled input signal is to be used, the integration interval is determined by the sampling interval. In this case the proposed method runs about 5 times faster than the Runge-Kutta method and both are to be preferred to the Trapezoidal Rule.

CONCLUSIONS

It has been demonstrated that an integration formula familiar in numerical analysis can be successfully applied to obtain rapid time response analysis of linear systems. A comparison with algorithms in present use suggests an improvement in speed in all cases except where the coarsest results are acceptable.

A problem similar to that stated in the introduction is that of computer-aided network design. The principal difference is that for general linear networks the cascade representation is not easily obtained. Nonlinear networks do not of themselves pose extra difficulties since they can be removed as memoryless elements from an otherwise linear network [5] . Available programs in this area use almost exclusively the Runge-Kutta or Trapezoidal Rule. An order of magnitude increase in the speed of such programs may be expected if the suggested formula can be applied in this area.

ACKNOWLEDGMENT

The author wishes to thank the Deutches Rechenzentrum of Darmstadt, Germany and the Institut für Mathematische Maschinen of the University of Erlangen-Nürnberg and its director Prof. Dr. W. Händler for making available time on the two computers mentioned in this report.

REFERENCES

[1] W. Kuntz and H. W. Schüssler, "Zur numerischen Berechnung der Ausgangsfunktion von Netzwerken mit Hilfe der Z - Transformation," NTZ, Vol. 3, 1966, pp. 169-172.

[2] A. Ralston and H. S. Wilf, Eds., Mathematical Methods for Digital Computers. New York: Wiley, 1960, p. 117.

[3] R. W. Hamming, Numerical Methods for Scientists and Engineers. New York: McGraw-Hill, 1962, pp. 128-130.

[4] J. Jess and H. W. Schüssler, "On the Design of Pulse-Forming Networks," I.E.E.E. Trans. on Circuit Theory, CT-12 (September 1965), pp. 393-399.

[5] R. H. Allen and C. Pottle, "High-speed Digital Computer Analysis of Nonlinear and Time-Varying Electronic Circuits," Proc. Fifth Allerton Conference, Monticello, Illinois, 1967.

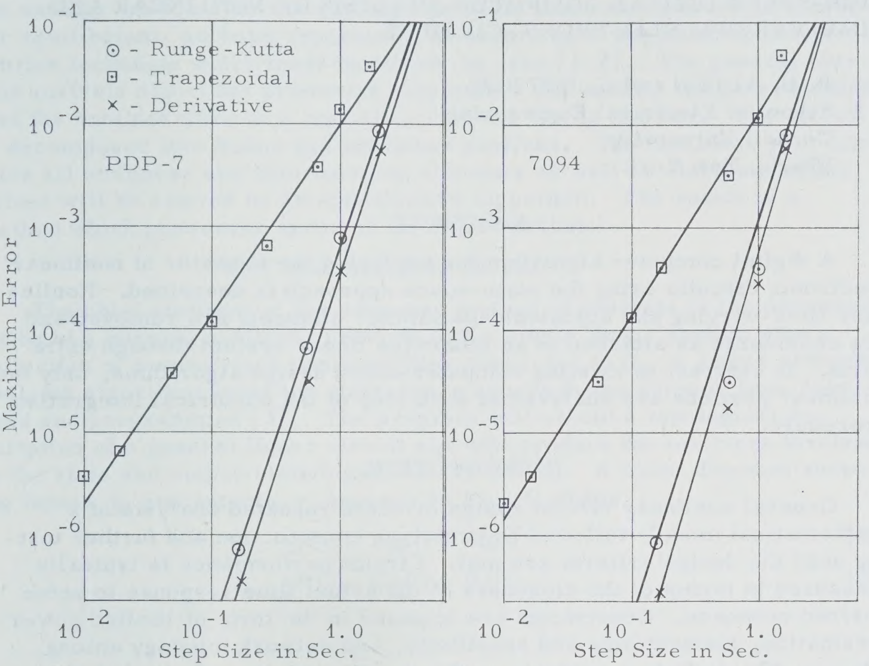


Fig. 4. Error vs. Step Size for Three Methods and Two Computers.

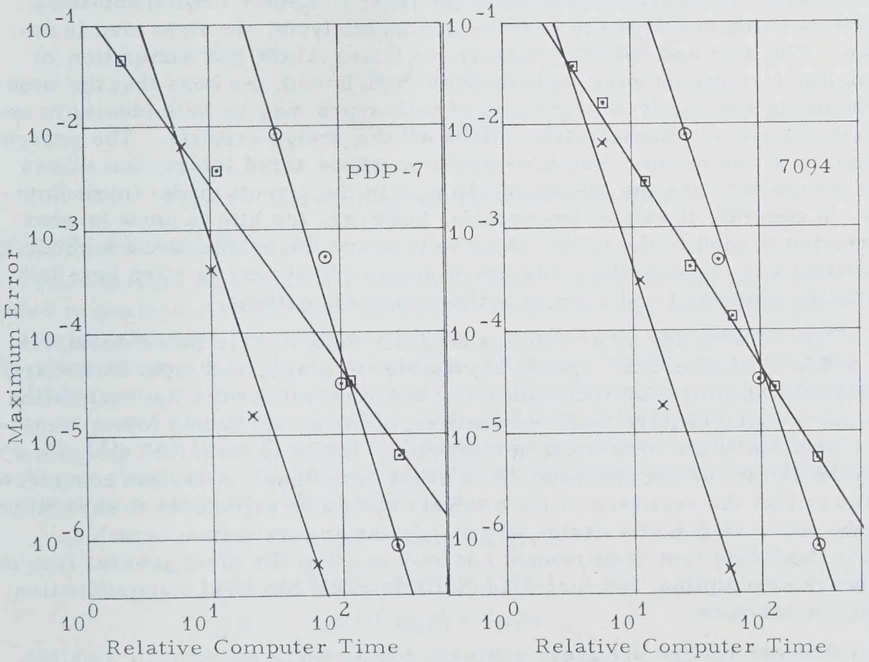


Fig. 5. Computer Time for Three Methods and Two Computers.

HIGH-SPEED DIGITAL COMPUTER ANALYSIS OF NONLINEAR AND TIME-VARYING ELECTRONIC CIRCUITS

R. H. ALLEN and C. POTTLE
School of Electrical Engineering
Cornell University
Ithaca, New York

ABSTRACT

A digital computer algorithm for analyzing the behavior of nonlinear electronic circuits using the state-space approach is described. Nonlinear, time-varying and adjustable-parameter elements are removed and are considered as attached to an otherwise linear system through extra ports. In contrast to existing computer-aided design algorithms, only the nonlinear portions are analyzed at each step of the numerical integration procedure.

INTRODUCTION

General nonlinear circuit design involves repeated analysis of a mathematical model, followed by prototype construction and further testing until the design criteria are met. Circuit performance is typically measured in terms of the closeness of the actual time response to some desired response. Constraints are imposed in the form of limited power dissipation, element size and sensitivity, and network topology among others. The digital computer now plays an increasing role in this design procedure. Numerical algorithms and their computer implementations are now being developed to perform model analysis, the first step in design. The size and speed of modern machines allow the simulation of complex circuits by very sophisticated models with the hope that the time-consuming and costly construction of prototypes may be held ideally to one realization which immediately fulfills all the design criteria. The present trend is to use on-line computer systems whose rapid interaction allows the designer to see the effects of changes in the circuit model immediately. In general, it will be impossible, however, for him to know in what direction to modify the model so as to improve the performance without violating any constraints. The job of model adjustment is often best left up to the computer and a set of optimization algorithms.

The criteria for an acceptable analysis algorithm in this context are, in order of importance: speed, adjustable accuracy, and input flexibility. All useful optimization techniques are based on successive approximation methods which require many evaluations of the performance index, usually a function of the network time response. Speed is therefore the prime consideration for the analysis phase of the technique. A second consideration is that the accuracy of the analysis should be adjustable to save time in the early stages of a design where a lower accuracy is acceptable. Input flexibility is a requirement not only to allow the most general type of network description, but also especially to allow for good communication with the machine.

Present on-line nonlinear analysis routines are enormously flexible but usually are too slow for repeated analysis since the network is solved for equilibrium, perhaps repeatedly, at each step of the numerical integration technique which must inevitably be used [1, 2]. The general circuit analysis algorithm presented here makes the usually valid assumption that the nonlinearities in a typical circuit are few in number. The circuit is decomposed into linear and nonlinear portions. The nonlinear part contains all nonlinear and time-varying elements as well as elements whose values will be altered by an optimization algorithm. The result is a method which possesses speed as its chief attribute.

SYSTEM DECOMPOSITION

The chief attribute of the proposed method of analysis is the decomposition of a general nonlinear network into a single linear system plus a collection of memoryless nonlinearities connected to it. A linear network analysis algorithm using the state-space approach has already been developed and programmed [3]. The program will accept a topological description of a general linear circuit and will produce the matrices involved in the state and output-state equations (1) and (2). A block diagram showing inputs $\underline{u}$ and outputs $\underline{y}$ appears in Fig. 1 where

$$\dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \tag{1}$$

$$\underline{y}(t) = C\underline{x}(t) + D\underline{u}(t) \quad . \tag{2}$$

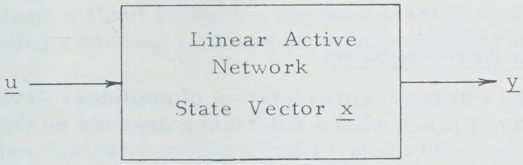


Fig. 1. Block Diagram of Linear System.

Nonlinear Element Removal

The analysis algorithm for invariant linear networks has been extended to analyze circuits with nonlinear, time-varying, and adjustable-parameter elements [4]. These varying elements are removed from the linear circuit by adding to the network a set of new inputs $\underline{w}$ and outputs $\underline{z}$ which represent the controlled and controlling quantities, respectively, of the removed elements. The new state and output-state equations are now (3), (4), and (5). The matrices involved are obtained directly from the linear algorithm. Equation (6) describes the behavior of the additional elements as a set of algebraic equations:

$$\dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) + E\underline{w}(t) \tag{3}$$

$$\underline{y}(t) = C\underline{x}(t) + D\underline{u}(t) + F\underline{w}(t) \tag{4}$$

$$\underline{z}(t) = G\underline{x}(t) + H\underline{u}(t) + J\underline{w}(t) \tag{5}$$

$$\underline{w}(t) = \underline{f}(\underline{z}(t), t) \quad . \tag{6}$$

It is important to note that the entire state vector appears in the linear portion of the network. The nonlinearities are described by purely algebraic (instead of differential) equations. Fig. 2 shows the varying part of the circuit separated from the linear part.

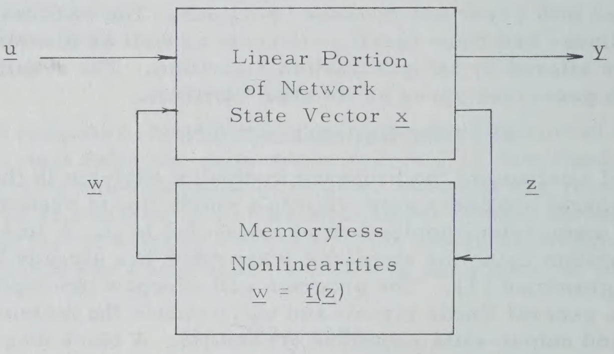


Fig. 2. Nonlinear System with Varying Elements Removed.

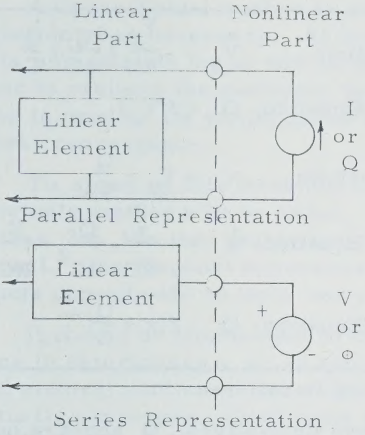
It is possible in the above framework to include nonlinear energy storage elements as well as memoryless nonlinearities. Nonlinear R, L, C and mutual elements as well as varying dependent and independent voltage and current sources are allowable. Other devices can be modeled by these elements.

Nonlinear Element Representation

To examine the internal representation of nonlinear devices, consider a nonlinear inductive branch whose inductance depends on the current through it.

$$v(t) = \frac{d\phi}{dt} = \frac{d(L(i)i)}{dt} = i\,\frac{dL(i)}{dt} + L(i)\,\frac{di}{dt} = \left(L + i\,\frac{dL}{di}\right)\frac{di}{dt} \quad . \tag{7}$$

To eliminate the need for derivatives of L, the basic definition of an inductive branch in terms of a current-flux relationship $\phi = f(i)$ is clearly more suitable than $\phi = L(i)i$. The assumption made here is that nonlinear capacitors are controlled by charge or voltage, while nonlinear inductors are controlled by current or flux. Correspondingly, flux and charge are added to the linear algorithm as allowable input and output quantities. Since the state variables of the linear model are voltages and currents, the state equations are written in terms of voltages and currents. Flux and charge inputs will thus appear as derivatives in the state equations which can be removed by a suitable transformation of the state vector [3]. These sources will have the same effect on the linear circuit as external independent voltage and current sources; namely, they will cause the loss of state variables if they enter into cutsets of inductors and current (or charge) sources or into loops of capacitors and voltage (or flux) sources. The internal representation of nonlinear and adjustable-parameter R, L, and C branches are therefore chosen from the set shown in Fig. 3.



Element Type	Controlled Variable w	Controlling Variable z
R	I	V
L	I	φ
C	Q	V
R	V	I
L	φ	I
C	V	Q

Fig. 3. Nonlinear Element Representation.

Every nonlinear element has a linear part removed from it which appears in the linear network. The most important reason for this is that for LC type elements, a state variable must be created in the linear portion of the circuit. The linear element guarantees its appearance. Furthermore the controlled sources without a passive element present could cause cutsets of current defining elements or loops of voltage defining elements to appear.

Restrictions

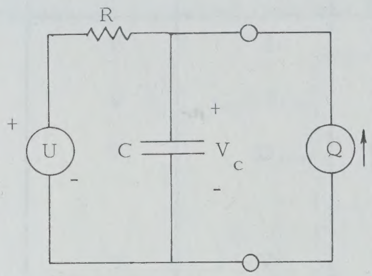
The question naturally arises whether it is always possible to obtain a network representation in terms of equations (3), (4), (5), and (6). In particular, when excess state variables are being removed from the state equation due to cutsets of current-defining elements or loops of voltage-defining elements, it is possible for derivatives of input quantities to arise both in the state equation (3) and control equation (5) leading eventually to an implicit differential equation (6). It can be shown, however, that for "proper" linear systems this situation cannot arise, since either the sources do not appear in the equation because of the nonlinear branch representation, or all source derivatives can be removed from both equations by a suitable transformation of the state vector. Only when the presence of carefully "tuned" active elements causes a reduction in the dimension of the state vector is it possible to obtain these derivatives. Such networks are possible for the linear case (e.g., an NIC), but are quite definitely pathological in the nonlinear case.

The Possibility of Iteration

Equation (6) defining the nonlinearities can in some cases become a set of implicit nonlinear algebraic equations involving $\underline{w}$ on both sides, depending on the form of the matrix J :

$$\underline{w} = \underline{f}(G\underline{x} + H\underline{u} + J\underline{w}) = \underline{f}(\underline{x}, \underline{u}, \underline{w}) \quad . \qquad (8)$$

As an example, consider the simple nonlinear circuit shown in Fig. 4.



State Equation: $\dot{V}_c = -\frac{V_c}{RC} + \frac{U}{RC} + \frac{\dot{Q}}{C}$

Nonlinear Equation: $Q = f(V_c)$

Transformation: $x = V_c - \frac{Q}{C}$

New State Equation: $\dot{x} = -\frac{x}{RC} - \frac{Q}{RC^2} + \frac{U}{RC}$

Nonlinear Equation: $Q = f(x + \frac{Q}{C})$

Fig. 4. Example Requiring Iteration.

After a state vector transformation to remove derivatives, Q appears on both sides of the nonlinear equation. In general, then, when the numerical integration procedure solving the state equation requires $\dot{\underline{x}}$, an iterative scheme must be used to solve the nonlinear equations for $\underline{w}$. Since the number of nonlinearities is assumed small, these iterations, which involve only the nonlinear functions themselves, should not be costly.

NUMERICAL CALCULATION OF NETWORK TIME BEHAVIOR

The original electronic circuit can be represented in terms of its state, output, control, and nonlinear equations (3), (4), (5) and (6). The problem now is to solve the state equation numerically to obtain the state as a function of time. The computer approach is to convert the differential equation (3) into a difference equation (9) whose solution approximates the true solution at a sequence of evenly spaced time points.

$$\int_{t_n}^{t_{n+1}} \dot{\underline{x}}\, dt = \int_{t_n}^{t_{n+1}} (A\underline{x} + B\underline{u} + E\underline{w})\, dt$$

or

$$\underline{x}(t_{n+1}) = \underline{x}(t_n) + \text{approx.}\left[\int_{t_n}^{t_{n+1}} (A\underline{x} + B\underline{u} + E\underline{w})\, dt\right] \tag{9}$$

The evaluation of the difference equation (9) is required once per step in time, although the approximation to the integral may require more than one evaluation of the integrand depending on the method used. For each derivative evaluation the inputs $\underline{u}$ and $\underline{w}$ must be available. The elements of $\underline{u}$ are available as input signals; $\underline{w}$ is the output from the nonlinearities whose calculation may require iteration. It can thus be seen that to calculate the network time response, two numerical procedures are necessary. At each step in time a set of nonlinear algebraic equations must be solved for $\underline{w}$. These values are then used to calculate a derivative vector $\dot{\underline{x}}$ enabling a differential equation solving algorithm to update the state vector $\underline{x}(t)$.

Nonlinear Equation Algorithm

A Gauss-Seidel method is currently used to solve the nonlinear equation when necessary. As soon as a new component w_i is calculated, it is immediately put to use so that the latest available information is used to evaluate the nonlinear functions. The solution at the previous step is used as the initial guess at the next step. This type of method has linear convergence.

To speed up this iteration a quadratically convergent modified Newton's method will be added. This method, due to Dennis [5], requires 2 n function evaluations by approximating the Jacobion by its diagonal. The diagonal approximation is appropriate since most nonlinear inputs depend only on their own control.

It should be emphasized at this point that very little work has been done in determining a set of sufficient conditions on the network topology and element nonlinearities which guarantee that this iteration process actually converges. Conditions such as those imposed by Katzenelson [1] and Dertouzos [2] would seem to apply here also. It is hoped, however, that somewhat weaker conditions may be found.

Differential Equation Algorithm

The current computer implementation of the nonlinear network analysis technique makes use of either the Runge-Kutta or Trapezoidal Rule integration techniques. The Trapezoidal Rule, equation (10), is fast because the derivative $\dot{\underline{x}}$ is required only once per step.

$$\underline{x}(t_{n+1}) = \underline{x}(t_n) + \frac{t_{n+1} - t_n}{2}\Big(\dot{\underline{x}}(t_{n+1}) + \dot{\underline{x}}(t_n)\Big) \quad . \tag{10}$$

The method requires little storage and allows the time step to be altered with ease. Its principal disadvantage is its second order accuracy, requiring many small steps to be taken.

The Runge-Kutta method requires 4 derivative evaluations per step, but has fourth order accuracy. Because no information is required from previous steps, the method is self-starting and allows easy alterations in step size. Because of the difference in order between these two methods, there always exists an error level above which it pays to use the Trapezoidal Rule and below which, Runge-Kutta. It is hoped that other faster fourth order methods will prove feasible for nonlinear analysis [6].

COMPUTER IMPLEMENTATION

This nonlinear network analysis algorithm is an extension of an already existing linear algorithm [3], for which a user's manual exists [7]. Additions and modifications were made to include the features necessary for analysis of nonlinear, time-varying, and adjustable-parameter circuits [4]. The implementation of ANALYSIS-NONLINEAR is written entirely in the FORTRAN language so that it is usable by all digital computer installations. The routines are laid out in modular form so that portions such as the input or integration routines can be easily changed.

Nonlinear and time-varying element values may be described to the computer in three different ways. If the element is described as a table of data, the user has the option of doing piecewise linear or quadratic interpolation in the table. If the element value can be described as a polynomial, only the coefficients of this polynomial need be entered. Finally the program includes a skeleton FORTRAN subrouting to which statements of any sort can be added to describe the nonlinearity. The subroutine can then be compiled immediately prior to execution.

The input to and output from the computer in analyzing the example described in the next section is shown in Fig. 7.

EXAMPLE

One use made of the nonlinear analysis program has been a study of a microwave Gunn (LSA) Oscillator [8]. The purpose of this investigation was to establish simple models for the microwave circuit followed by adjustment of load, capacitive susceptance and pulse magnitude to produce a maximum efficiency of power transfer. The voltage pulse rise time was varied to optimize the starting transients.

The model in Fig. 5 is a lumped-element approximation of the distributed circuit. The time-varying voltage source (1) represents the applied pulse, while the voltage-controlled current source (2) simulates the Gallium Arsenide crystal. The perfect transformer represents the coupling of the waveguide load to the resonant circuit. The load resistor (3) is removed from the linear circuit also, as its value is to be adjusted without recalculating state equations.

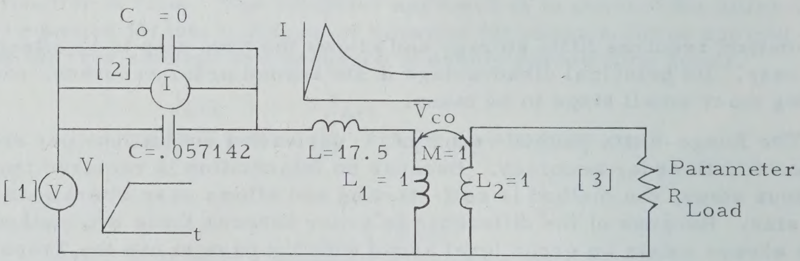


Fig. 5. A Model of a Gunn (LSA) Oscillator.

A sample of the computer input and output appears in Fig. 7. The three elements removed are defined in the FORTRAN subroutine NLFUNC. Scaling, integration and parameter information follow the circuit description. The output summarizes the circuit description, lists the state, output and control equations, and tabulates and plots the time response. Fig. 6 shows how the efficiency of power transfer depends on the circuit parameters. From these and similar plots, optimum capacitive susceptances, loads, and bias voltages may be determined.

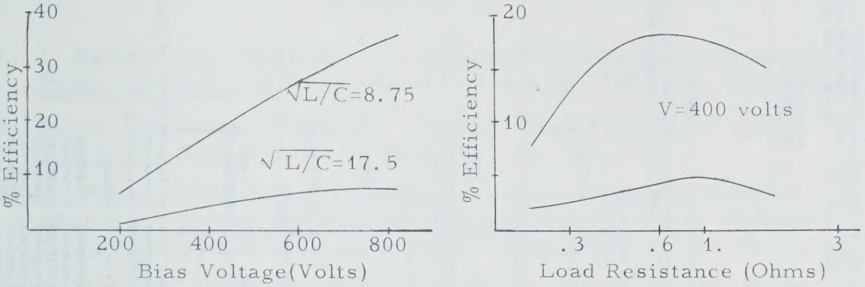


Fig. 6. Power Transfer Efficiency in Terms of Circuit Parameters.

CONCLUSION

This algorithm for the analysis of nonlinear, time-varying and adjustable-parameter networks has been designed to provide the following features:

1) The method should possess a speed edge over methods not taking advantage of portions of the network being linear.

2) The full power of the FORTRAN language is available for describing nonlinearities.

3) The class of circuits which can be analyzed is very general.

4) The FORTRAN implementation is machine independent.

Future Extensions

1) Since the parameter-adjusting capability has been included, the method should be imbedded in an optimization algorithm which uses response information to determine improved design parameters.

2) A search should be made for sufficient conditions on element nonlinearities which guarantee convergence of the numerical algorithms involved.

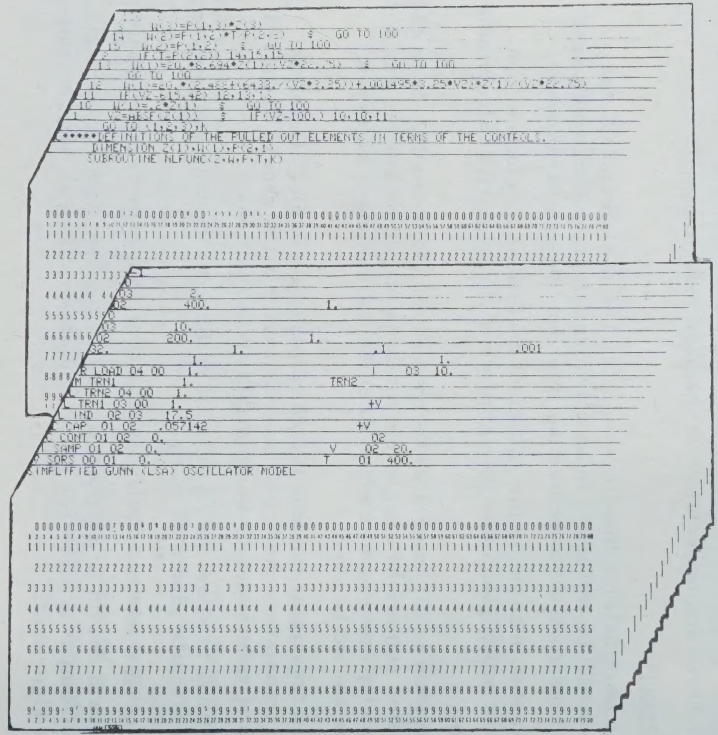
3) Improvements in all numerical techniques should be made. In particular, faster numerical integration methods should be studied and a quadratically convergent method to solve the nonlinear algebraic equations should be used.

4) Diode and transistor models should be made available. Possibly common transistor, diode and tube characteristics should be stored on disc and referenced by commercial identifications.

5) Modifications to allow the addition of multiport elements such as integrated circuits should be made.

6) Input and output power should be made available as program outputs.

7) The program should be adopted for interactive use with display console input-output to allow for more immediate dialogue with the optimization algorithm.



```
3     W(3)=P(1,3)*Z(3)
14    W(2)=P(1,2)*T/P(2,2)     $   GO TO 100
15    W(2)=P(1,2)   $   GO TO 100
2     IF(T-P(2,2)) 14,15,15
13    W(1)=20.*8.694*Z(1)/(V2*22.75)    $   GO TO 100
      GO TO 100
12    W(1)=20.*(2.488+(6433./(V2*3.25))+.001495*3.25*V2)*Z(1)/(V2*22.75)
11    IF(V2-615.42) 12,13,13
10    W(1)=.2*Z(1)    $   GO TO 100
1     V2=ABSF(Z(1))   $   IF(V2-100.) 10,10,11
      GO TO (1,2,3),K
*****DEFINITIONS OF THE PULLED OUT ELEMENTS IN TERMS OF THE CONTROLS.
      DIMENSION Z(1),W(1),P(2,1)
      SUBROUTINE NLFUNC(Z,W,P,T,K)
```

```
0
03          2.
02          400.                  1.
0
03          10.
02          200.                1.
32.                 1.                    .1                    .001
                  1.                                  1.
R LOAD 04 00    1.                      I     03   10.
M TRN1          1.                TRN2
L TRN2 04 00    1.
L TRN1 03 00    1.                     +V
L IND  02 03    17.5
C CAP  01 02    .057142              +V
C CONT 01 02    0.                     02
I SAMP 01 02    0.              V      02   20.
V SORS 00 01    0.              T      01   400.
SIMPLIFIED GUNN (LSA) OSCILLATOR MODEL
```

Figure 7. Computer Input and Output, for Example.
Above: FORTRAN Subroutine Defining the Removed Branches in Terms of Their Control.
Below: Input Data Describing Circuit Elements.
Right : Sample Output, for Example.

```
SIMPLIFIED GUNN (LSA) OSCILLATOR MODEL

 THE GRAPH OF THIS NETWORK IS DESCRIBED BY THE FOLLOWING BRANCHES
 BRANCH     NODE    ELEMENT     CONTROL CON-   OUT- EXT   EXT. NORM DATA NO.
TYPE NAME  NOS.     VALUE       MUTUAL TROL    PUT  NO.     VALUE    TYPE COF

  V  SORS  0  1            0           T            1  4.0000E+02
  I  SAMP  1  2            0           V            2  2.0000E+01
  C  CONT  1  2            0                        2
  C  CAP   1  2  5.7142E-02                 +V
  L  IND   2  3  1.7500E+01
  L  TRN1  3  0  1.0000E+00                 +V
  L  TRN2  4  0  1.0000E+00
  M  TRN1        1.0000E+00   TRN2
  R  LOAD  4  0  1.0000E+00            I            3  1.0000E+01

 THIS NETWORK HAS BEEN SCALED FOR COMPUTATION BY THE FOLLOWING FACTORS
 FREQUENCY 1.0000000E+00 RADIANS/SEC.      IMPEDANCE 1.0000000E+00 OHMS

      THE (SCALED) ENTRIES OF THE AO MATRIX ARE
  STATE
VARIABLES
              C CAP             L TRN1           L TRN2
  C CAP               0  1.7500263E+01                0
  L TRN1  -5.7142857E-02              0                0
  L TRN2   5.7142857E-02              0                0

      THE (SCALED) ENTRIES OF THE BO MATRIX ARE
  STATE
VARIABLES
              I SAMP            V SORS           V LOAD
  C CAP   -1.7500263E+01              0                0
  L TRN1              0 -5.7142857E-02 -5.7142857E-02
  L TRN2              0  5.7142857E-02  1.0571429E+00

      THE (SCALED) ENTRIES OF THE CO MATRIX ARE
 OUTPUT
VARIABLES
              C CAP             L TRN1           L TRN2
  V TRN1   5.8207661E-11              0                0
  V CAP    1.0000000E+00              0                0
  V CONT   1.0000000E+00              0                0
  I LOAD              0              0 -1.0000000E+00

      THE (SCALED) ENTRIES OF THE DO MATRIX ARE
 OUTPUT
VARIABLES
              I SAMP            V SORS           V LOAD
  V TRN1              0  5.8207661E-11  1.0000000E+00
  V CAP               0              0                0
  V CONT              0              0                0
  I LOAD              0              0                0

 TIME RESPONSE 0 TO 3.20E+01 SECS. MAX TIME INC 5.0E-02 OUTPUT ** BRANCH
  PARAMETERS FOR NONLINEARITY  2  2.000E+02  1.000E+00
  PARAMETERS FOR NONLINEARITY  3  1.000E+01          0

  TIME       V**CAP      V**TRN1      P**IN       P**OUT

  1.00E+00 -1.877E+00 -1.155E-01  2.946E-01  1.127E-01
  2.00E+00 -1.025E+01 -2.137E+00  1.889E+01  7.577E-01
  3.00E+00 -2.405E+01 -2.742E+00  1.521E+02  2.031E+00
  4.00E+00 -4.042E+01 -3.213E+00  6.018E+02  3.893E+00
  5.00E+00 -5.726E+01 -3.673E+00  1.591E+03  6.345E+00
  6.00E+00 -7.618E+01 -4.016E+00  3.425E+03  9.369E+00
  7.00E+00 -9.         -4.270E+00  6.490E+          E+01
```

REFERENCES

[1] J. Katzenelson, "AEDNET: A Simulator for Nonlinear Networks," Proc. I.E.E.E., Vol. 54, No. 11, November 1966, pp. 1536-1552.

[2] M. L. Dertouzos, "CIRCAL: On-line Circuit Design," Proc. I.E.E.E., Vol. 55, No. 5, May 1967, pp. 637-654.

[3] C. Pottle, "State Space Techniques for General Active Network Analysis," System Analysis by Digital Computer (F. F. Kuo and J. F. Kaiser, Editors). New York: Wiley, 1966, pp. 59-98.

[4] R. H. Allen, "State Variable Approach to the Numerical Solution of Nonlinear Systems," M. S. Thesis, Cornell University, Ithaca, New York, September 1966.

[5] J. Dennis, "Variations on Newton's Method," Ph. D. Thesis, University of Utah, Salt Lake City, Utah, 1966, p. 30.

[6] C. Pottle, "Rapid Computer Time Response Calculation for Systems with Arbitrary Input Signals," Proc. Fifth Allerton Conference, Monticello, Illinois, 1967.

[7] C. Pottle, "Comprehensive Active Network Analysis by Digital Computer - A State-Space Approach," Research Report EERL 59, Cornell University, Ithaca, New York, September 1966, Appendix: ANALYSIS.

[8] W. K. Kennedy, L. F. Eastman, and R. J. Gilbert, "LSA Operation of Long Bulk Ga As Oscillators," I.E.E.E. Trans. on Electronic Devices, ED-14, No. 9, 1967.

TRIDIAGONALIZATION AND RELATED MATRIX AND NETWORK ALGORITHMS FOR A POLYNOMIAL SPACE

THOMAS G. MARSHALL, JR.
Chalmers University of Technology *
Gothenburg, Sweden

ABSTRACT

In this paper some familiar algorithms of network and matrix theory are generalized so that the polynomials which appear are represented in any of several different ways. Particular examples of such representations are the Lagrange and Newton interpolation polynomials and some additional representations, to be defined, which are derived from a tridiagonal matrix.

The polynomials are considered as elements of a vector space with a linear transformation replacing the ring operation of polynomial multiplication. This description leads naturally to the use of matrices and avoids the determinants of interpolation theory. In this description the continued fraction expansion and tridiagonalization algorithms appear as special cases of a more general algorithm, and well known results pertaining to the former are extended to the latter. The main distinction between the two is seen to be one of representation. Comparisons with state-variable and classical network descriptions are made.

Computational experiments are described. Significant improvements in computational accuracy were obtained with certain choices of representations.

INTRODUCTION

It is often observed that a ladder network's response can be accurately calculated at a specific frequency using a few digits, whereas the determination of its response function as a ratio of polynomials requires many more digits for comparable accuracy. Similar observations hold with respect to the evaluation of a matrix' characteristic polynomial versus the determination of its coefficients. These discrepancies in digital requirements suggest the use of interpolation polynomials and other representations for the network or matrix polynomials so that their coordinates are better functional values than polynomial coefficients.

Some network and matrix analysis and synthesis procedures will be generalized below to allow the polynomials which appear to be represented in a wide variety of ways. These procedures are all related to performing a continued fraction expansion, or its vector space analog, tridiagonalization of a matrix, as well as the inverse procedure of finding the network response or the matrix characteristic polynomial. Matrix tridiagonalization is valuable as an intermediate step in the solution of eigenvalue problems [1,2] and for the stability analysis of a matrix [3]. Furthermore, it has been shown [4,5] that the tridiagonalization of a matrix in normal form is equivalent to the synthesis of an RLC ladder network. Thus, there are many areas in which matrix tridiagonalization plays a role.

The desired additional representations are introduced in the foregoing procedures in a convenient manner, avoiding the familiar determinants of interpolation theory, by considering the polynomials to be elements of a vector space. The procedures retain their original forms, but the operation of polynomial multiplication is redefined for each representation; more precisely the ring operation of polynomial multiplication is replaced by a linear transformation. The matrix of this transformation has different forms appropriate to the representation. A tridiagonal matrix which approximates the solution defines a representation in which cancellation errors are small.

A brief statement of the tridiagonalization procedure will be made in the next section. The following section will discuss the applications, and succeeding sections will be devoted to the development of the algorithms in a

---

* On leave of absence from Radio Corporation of America.

polynomial space by evolution of well known algorithms in a ring of polynomials.

TRIDIAGONALIZATION OF A PAIR OF MATRICES

In the following sections algorithms which solve the following problem will be developed. Given matrices $\underline{A}$ and $\underline{B}$ with real entries of order n, find tridiagonal matrices $\underline{T}$ and $\underline{T}^{-}$ with real entries satisfying

$$\underline{A} \sim \underline{T} \tag{1}$$

and

$$\underline{B} \sim \underline{T} \tag{2}$$

where

$$\underline{T} = \begin{bmatrix} -\delta_1 & -\chi_1 & & & \\ 1 & -\delta_2 & -\chi_2 & & \\ & 1 & \cdot & \cdot & \\ & & \cdot & \cdot & \cdot \\ & & & \cdot & \cdot & -\chi_{n-1,n} \\ & & & & 1 & -\delta_n \end{bmatrix}. \tag{3}$$

The following abreviated notation for a tridiagonal matrix, in which only the entries which in general are nonzero in each row are shown, will be employed:

$$\underline{T} = tdiag(-\delta_1\ -\chi_1,\ 1\ -\delta_2\ -\chi_2,\ \cdots,\ 1\ -\delta_n). \tag{4}$$

In this notation

$$\underline{T}^{-} = tdiag(+\delta_1\ -\chi_1,\ 1\ -\delta_2\ -\chi_2,\ \cdots,\ 1\ -\delta_n), \tag{5}$$

that is $\underline{T}^{-}$ differs from $\underline{T}$ only in the sign of the entry in location $1,1$.

The matrices $\underline{A}$ and $\underline{B}$ are, in general, assumed to be unit upper Hessenberg matrices, i.e. upper triangular matrices bordered by a subdiagonal of 1's. $\underline{T}$ and $\underline{T}^{-}$ are of this form. Any matrix similar to a companion matrix, a cyclic (or nonderogatory) matrix, can readily be put in this form [1,2]. The tridiagonalization procedure applies to the case where $\underline{A}$ and $\underline{B}$ are independently specified; but, most often, only one matrix $\underline{A}$ or $\underline{B}$ will be given, and the other will be derived from it in an appropriate manner for the application (synthesis, stability analysis, or eigenvalue problem).

The tridiagonalization of $\underline{A}$ and $\underline{B}$, Equations (1) and (2), is not performed directly. Instead a matrix $\underline{M}$, which determines the mode of representation, is employed, and a row vector $\underline{r}_0$ is determined, which satisfies

$$\underline{RMR}^{-1} = \underline{A} + \underline{e}_1'\underline{r}_0\underline{R}^{-1}. \tag{6}$$

The vector $\underline{r}_0$ is the coordinate vector of the characteristic polynomial of $\underline{A}$, $|\lambda\underline{I} - \underline{A}|$, in the representation defined by $\underline{M}$; the vector $\underline{e}_1$ is the first row vector of the unit matrix $\underline{I}$ and $\underline{e}_1'$ its transpose; and the rows of matrix $\underline{R}$, denoted $\underline{r}_i$, are coordinate vectors of principal minors of $|\lambda\underline{I} - \underline{A}|$. The algorithm for finding $\underline{r}_0$ will be seen to be a generalization of the familiar algorithm for finding the characteristic polynomial of a Hessenberg matrix or the response function of a ladder network by recursively determining polynomials of successively higher order.

The same algorithm is employed for finding the coordinate vector $\underline{s}_0$ of $|\lambda\underline{I} - \underline{B}|$ satisfying

$$\underline{SMS}^{-1} = \underline{B} + \underline{e}_1'\underline{s}_0\underline{S}^{-1}. \tag{7}$$

The tridiagonalization of $\underline{A}$ is completed by finding $\underline{T}$ satisfying the following equations,

$$\underline{YMY}^{-1} = \underline{T} + \underline{e}_1'\underline{y}_0\underline{Y}^{-1} \tag{8}$$

and

$$\underline{y}_0 = \underline{r}_0. \tag{9}$$

The tridiagonalization of $\underline{B}$ is automatically accomplished by the tridiagonalization of $\underline{A}$, and need not be explicitly performed, and the equations

$$\underline{YMY}^{-1} = \bar{\underline{T}} + \underline{e}_1'\bar{\underline{y}}_0\underline{Y}^{-1} \tag{10}$$

and

$$\bar{\underline{y}}_0 = \underline{s}_0 \tag{11}$$

are satisfied. The algorithm is successful if and only if $\delta_1 \neq 0$ and $\chi_i \neq 0$ for $i=1,\cdots,n-1$.

Equation (6), (7), (8), and (10) may be expressed in equivalent forms as illustrated below for Equation (8),

$$\underline{Y}(\underline{M} - \underline{Y}^{-1}\underline{e}_1'\underline{y}_0)\underline{Y}^{-1} = \underline{T}, \tag{12}$$

where $\underline{Y}^{-1}\underline{e}_1'$ is recognized to be the first column of $\underline{Y}^{-1}$. The equivalence is easily established by premultiplying the second term in Equation (8) by $\underline{YY}^{-1}$.

In the special case where $\underline{M}$ is chosen so that $\underline{M} = \underline{A}$, Equation (6) is satisfied trivially, $\underline{r}_0 = \underline{y}_0 \equiv 0$, and Equation (12) becomes $\underline{YAY}^{-1} = \underline{T}$. The procedure then is nearly equivalent to those of references [1,2] (with the addition of a matrix $\underline{B}$ which serves to specify the initial vector $\underline{y}_1$. An additional distinction, that will be described, is that a generalized method is introduced of forming certain column vectors, which are required in the execution of Equation (8)).

APPLICATIONS

Matrix Eigenvalue Problem

The tridiagonalization of a Hessenberg matrix, $\underline{A}$, is an important, but numerically critical, step in a recommended procedure for solving the eigenvalue problem for a nonsymmetric matrix [1,2]. The procedure of the preceding section provides a method for doing the tridiagonalization which has the added degree of freedom, as compared to the currently employed methods, of allowing the representation of the polynomials which (implicitly) appear in tridiagonalization to be specified. The matrix $\underline{B}$ could be an arbitrarily specified Hessenberg matrix of the same order as $\underline{A}$, which corresponds to an arbitrary selection of the initial vectors in the currently employed procedures, i.e. Lanczos' Algorithm and related algorithms. However, the hazards of an arbitrary choice can be avoided by deriving $\underline{B}$ from $\underline{A}$ in the manner that will be described for stability testing. For example, when all the eigenvalues of $\underline{A}$ are in the left half plane, it is assured that the tridiagonalization will be successful with the choice of $\underline{B}$ prescribed in the next section, whereas, an arbitrary choice may lead to premature termination of the algorithm. Another advantage of this particular choice of $\underline{B}$ is that fewer computations are required. A further advantage may exist. In the experiments to be described, there were fewer cancellation errors with this choice of $\underline{B}$ than with another choice, an observation which suggests that further such comparison experiments are warranted, in view of the fact that cancellation errors can cause severe loss of accuracy in this part of the eigenvalue problem.

Stability Testing

By analogy with physical systems, stability testing of a matrix will mean testing to see if all eigenvalues are in the left half plane. The number of eigenvalues in the left half plane of $\underline{A}$ can be determined by finding the tridiagonal matrix $\underline{T}$ similar to $\underline{A}$, if it exists, which has the property that

$$\delta_i = 0, \quad i = 2, \cdots, n. \tag{13}$$

See Reference [3].

Let the n eigenvalues of $\underline{A}$ be denoted $\lambda_i = \sigma_i + j\omega_i$ for $i = 1,\cdots, n$. Then, if $\underline{B}$ is chosen so that its eigenvalues, μ_i, appropriately ordered satisfy

$$\mu_i = -\sigma_i + j\omega_i \tag{14}$$

for $i = 1,\cdots, n$, the tridiagonalization procedure of the preceding section will give $\underline{T}$ satisfying Equation (13) when $\underline{T}$ exists. A matrix $\underline{B}$ satisfying Equation (14) can be constructed very simply from $\underline{A}$ merely by changing the signs of the entries in the main diagonal and in all diagonals above it, which are removed by an even number of diagonals, so that

$$b_{ij} = (-1)^{j-i+1} a_{ij}, \tag{15}$$

for $i = 1,\cdots,n$ and $j \geq i$. The following theorem is a statement of these conclusions for unit upper Hessenberg matrices, $\underline{A}$ and $\underline{B}$.

Theorem 1: When the tridiagonal matrix $\underline{T}$ exists satisfying $\underline{T} \sim \underline{A}$ and Equation (13), the tridiagonalization of Equations (6)-(11), with $\underline{B}$ specified by Equation (15), gives $\underline{T}$.

The proof of this theorem depends upon first establishing that Equation (14) holds, which can be established by induction for the rows of $\underline{R}$ and $\underline{S}$ and $\underline{r}_0$ and $\underline{s}_0$. The remainder of the theorem follows, as a result of the succeeding development, from well known results concerning the continued fraction expansion of the ratio of the even and the odd parts of a polynomial.

RLC Ladder Network Synthesis.

A lowpass *RLC* ladder network with all transmission zeros at infinity, which will be referred to as a ladder network, can be synthesized by tridiagonalization of a pair of matrices. If branch i is series, let its inductance be ε_i and its series resistance be ρ_i, and if branch i is shunt, let its capacitance be ε_i and its shunt conductance be ρ_i. Then, an n branch ladder network, with $e_i \neq 0$ for $i = 1, \cdots, n$ determines a unique tridiagonal matrix $\underline{T}$ with

$$\delta_i = \rho_i/\varepsilon_i \tag{16}$$

and

$$\chi_i = 1/\varepsilon_i\varepsilon_{i+1}, \tag{17}$$

and the eigenvalues of $\underline{T}$ are the natural frequencies of the network. For synthesis the converse relationship is needed, that if $\chi_i \neq 0$ for $i = 1, \cdots, n-1$, one ε_i may be arbitrarily specified (to determine the impedance level), and then the remaining ε_i and all the ρ_i are uniquely determined by $\underline{T}$ and Equations (16) and (17). Each such set of values determines a pair of dual ladder networks. It is seen that $\underline{T}$ determines a complete class of ladder networks, and that synthesis of $\underline{T}$ serves to synthesize this class.

The preceding tridiagonalization procedure solves the following realization problem. A reflection coefficient $r_{11}(\lambda)$ is given as the ratio of two monic polynomials of degree n,

$$r_{11}(\lambda) = b(\lambda)/a(\lambda). \tag{18}$$

Let $\underline{A}$ be any (unit upper Hessenberg) matrix such that $|\lambda\underline{I}-\underline{A}| = a(\lambda)$ and $\underline{B}$ any such matrix with $|\lambda\underline{I}-\underline{B}| = b(\lambda)$. Then the conditions for the success of the tridiagonalization procedures, $\delta_1 \neq 0$ and $\chi_i \neq 0$ for $i = 1,\cdots,n-1$, are sufficient to insure that a class of ladder networks which realize $r_{11}(\lambda)$ can be obtained from $\underline{T}$ and Equations (16) and (17). Furthermore, if a ladder network and its correspond matrix $\underline{T}$ exist with $\delta_1 = 0$, it is necessary that $a(\lambda) = b(\lambda)$. These considerations can be used to establish the following theorem.

Theorem 2: If a ladder network exists which realizes $r_{11}(\lambda)$, which is the ratio of $b(\lambda)$ and $a(\lambda)$ which are polynomials of the same degree satisfying $a(\lambda) \neq b(\lambda)$ but otherwise arbitrarily specified, then the ladder network can be synthesized by the described procedure.

It is seen that *RLC* ladder networks, possibly having negative elements, can be synthesized. The most common situation is when $a(\lambda)$ is derived from $b(\lambda)$, or vica versa, so that the ladder network is internally lossless, i.e.

$\delta_i = 0$ for $i = 2,\cdots,n-1$ and all χ_i are positive [5,6,7].

Some convenient choices for $\underline{A}$ and $\underline{B}$, whose characteristic polynomials are respectively $a(\lambda)$ and $b(\lambda)$, are as follows. If the roots of $a(\lambda)$ and $b(\lambda)$ are given (which is usually the starting point for the realization problem), $a(\lambda)$ and $b(\lambda)$ may be expressed as products of linear and quadratic factors with real coefficients, for example

$$a(\lambda) = (\lambda^2 + \delta_1\lambda + \beta_1)(\lambda + \alpha_3)(\lambda^2 + \alpha_4\lambda + \beta_4). \tag{19}$$

The corresponding matrix, $\underline{A}$, can then be

$$\underline{A} = tdiag(-\alpha_1\ -\beta_1,\ 1\ \ 0\ \ 0,\ 1\ -\alpha_3\ \ 0,\ 1\ -\alpha_4\ -\beta_4,\ 1\ \ 0). \tag{20}$$

The subscripts in Equation (19) are the row numbers in $\underline{A}$. $\underline{A}$ is block upper triangular, each block being the companion matrix of a factor of $a(\lambda)$, so its characteristic function is $a(\lambda)$. A second choice of $\underline{A}$ and $\underline{B}$, which is less convenient unless the coefficients of $a(\lambda)$ and $b(\lambda)$ are known, is to let them be the companion matrices of $a(\lambda)$ and $b(\lambda)$.

Perturbation Theory I

Consider the case when $\underline{M}$ is a matrix $\underline{T}_0$ which is an approximation to $\underline{T}$. By definition, $\underline{A} \sim \underline{T}$, so the vector $\underline{y}_0$ determined from Equations (6) and (9) is an approximation to the zero vector. In turn, the matrix $\underline{Y}$ is an approximation to the unit matrix $\underline{I}$. If $\underline{Y}$ is unit upper triangular, as it can be because $\underline{T}_0$ is unit upper Hessenberg, $\underline{Y}$ is different from $\underline{I}$ only above the main diagonal.

These observations suggest that the choice of $\underline{T}_0$ for $\underline{M}$, in preference to other choices such as $\underline{M} = \underline{A}$, may reduce the cancellation errors normally experienced in determining the row vectors of $\underline{Y}$, $\underline{y}_i$, because the vectors are then approximations to the unit vectors $\underline{e}_i$. The resulting matrix, denoted $\underline{T}_1$, can be substituted for $\underline{M}$ to get $\underline{T}_2$, and the procedure repeated iteratively. Thus an iterative procedure is defined which is an algorithm for improving upon an approximate solution of the tridiagonalization of a matrix $\underline{A}$. The cancellation errors in the $\underline{y}_i$ can be shown to become of second order subject to the above assumptions, so improvement might be expected as this occurs. The accuracy of the final $\underline{T}$ as a solution to Equation (1) is then limited by the cancellation errors in the computation of $\underline{r}_0$ and round-off errors. The computation of $\underline{s}_0$ in Equation (1) does not directly affect the accuracy of the solution of Equation (1) (recall that $\underline{B}$ can be arbitrary), but a corresponding improvement in the solution of Equation (2) would be expected to accompany that in Equation (1). Perturbation theory will be discussed again in a later section. Some experiments that have been performed with this and the preceding applications of the tridiagonalization procedure will be described in a succeeding section.

ALGORITHMS FOR POLYNOMIAL RINGS

The tridiagonalization procedure was seen to require two algorithms, one for determining the characteristic polynomials of $\underline{A}$ and $\underline{B}$, $a(\lambda)$ and $b(\lambda)$, and one for obtaining a tridiagonal matrix, $\underline{T}$ which has $a(\lambda)$ for its characteristic polynomial and $b(\lambda)$ as the characteristic polynomial of the associated tridiagonal matrix $\underline{T}^-$. These two algorithms will be stated below, firstly, employing the normal ring operations and, secondly, in a modified form for a quotient ring [9]. In this manner the algorithms in a vector space of polynomials may be obtained directly.

Polynomial Ring

It is assumed that unit upper Hessenberg matrices $\underline{A}$ and $\underline{B}$ are given. It will be convenient to give their characteristic polynomials the subscript 0, so that they become $a_0(\lambda)$ and $b_0(\lambda)$ respectively. The characteristic polynomials of $\underline{T}$ and $\underline{T}^-$ will be denoted $t_0(\lambda)$ and $t_0^-(\lambda)$, respectively. The natural representation for a monic polynomial, say $a_0(\lambda)$, will be defined as

$$a_0(\lambda) = \lambda^n + a_{01}\lambda^{n-1} + a_{02}\lambda^{n-2} + \cdots + a_{0n}. \tag{21}$$

It will be convenient to introduce the scalars α_{ij} which are the negatives of the entries a_{ij} in the upper triangle of $\underline{A}$

$$\alpha_{ij} = -a_{ij}, \qquad j \geq i. \tag{22}$$

The algorithm for determining $a_0(\lambda)$ in the natural representation will now be stated. A related polynomial, $\bar{a}_0(\lambda)$ is determined in Step 3 and will be discussed.

Algorithm 1: Given $\underline{A}$, find $\{a_{0j}\}$ and $\{\bar{a}_{0j}\}$.

```
1 - Initialization of loop
    a -   $a_n(\lambda) = 1$                                                        (23)
    b -   $a_{n-1}(\lambda) = \lambda + \alpha_{nn}$                                 (24)
2 - Loop for $i = n-1, \cdots, 1$
    a -   $a_{i-1}(\lambda) = (\lambda + \alpha_{ii})a_i(\lambda) + \Sigma_{j=i+1}^{n} \alpha_{ij} a_j(\lambda)$      (25)
3 -       $\bar{a}_0(\lambda) = (\lambda - \alpha_{11})a_1(\lambda) + \Sigma_{j=2}^{n} (-1)^j \alpha_{1j} a_j(\lambda)$  (26)
```

Steps 1 and 2 are the well known algorithm for finding the characteristic polynomial of a unit upper Hessenberg matrix.

The polynomial $\bar{a}_0(\lambda)$ obtained in Step 3 is the characteristic polynomial of a matrix, denoted $\underline{A}^-$, obtained from $\underline{A}$ by changing the signs of entries in the first row which are in odd numbered columns, i.e.

$$\bar{\alpha}_{1j} = (-1)^j \alpha_{1j}, \qquad j = 1, \cdots, n \tag{27}$$

All other entries of $\underline{\bar{A}}$ are identical to those of $\underline{A}$. The polynomial $\bar{a}_0(\lambda)$ is not normally used in the tridiagonalization procedure itself, but finds use as a test of the results, for then Algorithm 1 can be applied to the resulting $\underline{T}$, and the polynomials $a_0(\lambda)$ and $\bar{a}_0(\lambda)$ are calculated versions of $t_0(\lambda)$ and $\bar{t}_0(\lambda)$ to which they can be compared.

The second algorithm is as follows:

Algorithm 2: Given $\{t_{0j}\}$ and $\{\bar{t}_{0j}\}$, find $\underline{T}$.

```
1 - Initialization of loop
    a -   $\delta_1 t_1(\lambda) = (t_0(\lambda) - \bar{t}_0(\lambda))/2$                 (28)
    b -   $\delta_1 = (t_{01} - \bar{t}_{01})/2$                                         (29)
    c -   Terminate if $\delta_1 = 0$; otherwise find $t_1(\lambda)$.
    d -   $t_{n-1,n+1} = t_{n,n+1} \equiv 0$                                             (30)
2 - Loop for $i = 1, \cdots, n-1$
    a -   $\chi_i t_{i+1}(\lambda) = t_{i-1}(\lambda) - (\lambda + \delta_i)t_i(\lambda)$   (31)
    b -   $\chi_i = t_{i-1,i+1} - \delta_i t_{i,i+1} - t_{i,i+2}$                         (32)
    c -   Terminate if $\chi_i = 0$; otherwise find $t_{i+1}(\lambda)$.
    d -   $\delta_{i+1} = t_{i,i+1} - t_{i+1,i+2}$                                       (33)
```

The algorithm follows from the expressions

$$t_{i-1}(\lambda) = (\lambda + \delta_i)t_i(\lambda) + \chi_i t_{i+1}(\lambda) \tag{34}$$

and

$$\bar{t}_0(\lambda) = (\lambda - \delta_1)t_1(\lambda) + \chi_1 t_2(\lambda) \tag{35}$$

which are special cases of Equations (25) and (26) which apply when $\underline{A} = \underline{T}$. This algorithm is the inverse of Algorithm 1 applied to $\underline{T}$. The polynomials are continuants [8], i.e.

$$t_{i-1}(\lambda) = K(i,n), \tag{36}$$

and the algorithm determines $t_1(\lambda)$ and the continued fraction

$$\frac{t_0(\lambda)}{t_1(\lambda)} = \lambda+\delta_1 + \frac{\chi_1}{\lambda+\delta_2 +}\ \frac{\chi_2}{\lambda+\delta_3 +}\ \cdots\ \frac{\chi_{n-1}}{\lambda+\delta_n}, \tag{37}$$

when they exist.

QUOTIENT RING

A preliminary step in the development of the algorithms in a vector space is to alter the preceding algorithms so that polynomials of degree n do not occur. To do this all polynomials will be replaced by their remainders upon division by $m_0(\lambda)$, a monic polynomial of degree n. The resulting structure is a quotient ring. Corresponding to the polynomials $a_i(\lambda), b_i(\lambda), t_i(\lambda)$, and $\bar{t}_i(\lambda)$ are the remainder polynomials, denoted $r_i(\lambda), s_i(\lambda), y_i(\lambda)$ and $\bar{y}_i(\lambda)$. Because the former polynomials and $m_0(\lambda)$ are monic polynomials, the remainders are particularly simple as illustrated for the nth degree polynomials, $t_0(\lambda), \bar{t}_0(\lambda)$ and $\lambda t_1(\lambda)$

$$y_0(\lambda) = t_0(\lambda) - m_0(\lambda), \tag{38}$$

$$\bar{y}_0(\lambda) = \bar{t}_0(\lambda) - m_0(\lambda), \tag{39}$$

and

$$\lambda y_1(\lambda) = \lambda t_1(\lambda) - m_0(\lambda). \tag{40}$$

It also follows that

$$y_i(\lambda) = t_i(\lambda),\ i=1,\cdots,n. \tag{41}$$

Corresponding to Algorithms 1 and 2 are two algorithms in the quotient ring, Algorithms 3 and 4. Only the latter will explicitly stated.

Algorithm 4: Given $\{y_{0i}\}$ and $\{\bar{y}_{0i}\}$, find $\underline{T}$.

```
1 - Initialization of loop
    a - $\delta_1 y_1(\lambda) = (y_0(\lambda) - \bar{y}_0(\lambda))/2$          (42)
    b - $\delta_1$ = 1st coefficient of $\delta_1 y_1(\lambda)$          (43)
    c - Terminate if $\delta_1 = 0$; otherwise find $y_1(\lambda)$.
    d - $z_1(\lambda) = y_0(\lambda) - \lambda y_1(\lambda)$          (44)
2 - Loop for $i = 1,\cdots,n-1$
    a - $\chi_i y_{i+1}(\lambda) = z_i(\lambda) - \delta_i y_i(\lambda)$          (45)
    b - $\chi_i$ = $(i+1)$th coefficient of $\chi_i y_{i+1}(\lambda)$          (46)
    c - Terminate if $\chi_i = 0$; otherwise find $y_{i+1}(\lambda)$.
    d - $z_{i+1}(\lambda) = y_i(\lambda) - \lambda y_{i+1}(\lambda)$          (47)
    e - $\delta_{i+1}$ = $(i+1)$th coefficient of $z_{i+1}(\lambda)$          (48)
```

The polynomials $z_i(\lambda)$ were added here and explicit use of polynomial coefficients was omitted in anticipation of the succeeding development where the representation of the polynomials will be changed and their coefficients won't be directly available. The relationship between Algorithms 2 and 4 will be stated as a theorem.

Theorem 3: Algorithm 4, when initiated with $y_0(\lambda)$ and $\bar{y}_0(\lambda)$ of Equations (38) and (39), yields the same δ_i and χ_i as Algorithm 2, and Equation (41) is satisfied.

The proof follows directly from Equations (38) - (40).

A POLYNOMIAL SPACE

The Transformation M

Let P_n be the vector space [9,10] of polynomials of degree $<n$, with

typical element $p=p(\lambda)$. In the natural representation

$$p = \alpha_1\lambda^{n-1} + \alpha_2\lambda^{n-2} + \cdots + \alpha_n. \tag{49}$$

The polynomials in the quotient ring previously considered, with polynomial addition as normally defined, are in P_n. The product of two polynomials is not defined in P_n, but this operation will not be required.

Let a transformation M be defined using the natural basis elements, λ^i, of P_n,

$$\lambda^i M = \lambda^{i+1},\ i = 0,\cdots,n-1 \tag{50}$$

and

$$\lambda^{n-1}M = \lambda^n - m_0(\lambda). \tag{51}$$

The image of this transformation, pM, is the same as the product of λ times p in the quotient ring.

Theorem 4: M is a cyclic transformation.
Proof: If there is a vector p such that $\{p,pM,\cdots,pM^{n-1}\}$ is linearly independent, the transformation M is said to be cyclic (or nonderogatory) [9]. In this case, let $p = 1$, and the linear independence of the set follows from the definition of M.

The Natural Linear Functionals.

It was seen in Algorithm 4 that it was necessary to find the leading coefficient of a polynomial. The operation of selecting the jth coefficient of p is the linear functional f_j defined by

$$f_j(p) = \alpha_j. \tag{52}$$

This set of n functionals is a basis for the dual space of P_n, here denoted P'_n. It is seen to be biorthonormal to the natural basis of P_n, i.e.

$$f_j(\lambda^{n-i}) = \delta_{ij}, \tag{53}$$

and will accordingly be referred to as the natural basis for P'_n. An arbitrary linear functional, f, will be denoted, in the natural representation,

$$f = \beta_1 f_1 + \beta_2 f_2 + \cdots + \beta_n f_n. \tag{54}$$

In order to perform the tridiagonalization procedure, it will be seen to be sufficient to know the 1st natural functional f_1 in a representation which is the dual of that employed for the polynomials.

The Algorithms for a Polynomial Space

The preceding observations permit a restatement of Algorithm 3 for the polynomial space that was introduced. The unit transformation is denoted I.

Algorithm 5: Given $\underline{A}$, find r_0 and $\bar{r}_0$.

```
1 - Initialization of loop
    a -   $r_n = \lambda^0$                                                  (55)
    b -   $r_{n-1} = r_n(M + \alpha_{nn}I)$                                  (56)
2 - Loop for $i = n-1,\cdots,1$
    a -   $r_{i-1} = r_i(M+\alpha_{ii}I) + \Sigma_{k=i+1}^{n}\alpha_{ik}r_k$      (57)
3 -       $\bar{r}_0 = r_1(M-\alpha_{11}I) + \Sigma_{k=2}^{n}(-1)^k\alpha_{1k}r_k$  (58)
```

Two algorithms will be referred to as equivalent if, given the same input, they yield the same output.

Theorem 5: Algorithms 3 and 5 are equivalent.
The proof follows directly from the definitions of M and I by considering the r_i in the natural representation.

Algorithm 4 requires that the lead coefficients of polynomials of degrees 0 to $n-1$ be determined. The natural functionals suffice for this, but in view of the fact that it is only the lead coefficient that need be determined, a more general, and more convenient, set of functionals, denoted $\{h_i\}$, will be introduced. Let $h_1 = f_1$, the first natural functional. For p of degree $n-i$, pM^{i-1} is of degree $n-1$, and $f_1(pM^{i-1}) = \alpha_i$, its lead coefficient. But, by the bilinear identity, $f(pM) = fM'(p)$, the functional $f_1M'^{i-1}$ yields α_i, and $f_1M'^{j-1}$, for $j=1,\cdots,i-1$, yields the higher order coefficients which are zero. Define h_{i+1} recursively as

$$h_{i+1} = h_iM' + \Sigma_{k=1}^{i}\gamma_{ki}h_k, \tag{59}$$

with $h_1 = f_1$ and arbitrary γ_{ki}. The following "lead coefficient" lemma is established.

Lemma: The lead coefficient, α_i, of a polynomial p of degree $n-i$, for $i = 1,\cdots,n$, is $h_i(p)$.

The arbitrary nature of the γ_{ki} permits h_i to be selected, for a given representation, so as to minimize the computations. Specific choices will be discussed in the next section. The counterpart to Algorithm 4 will now be stated.

Algorithm 6: Given y_0 and $\bar{y}_0$, find $\underline{T}$.

```
1 - Initialization of loop
    a -   $h_1$ = 1st natural functional                 (60)
    b -   $\delta_1 y_1 = (y_0 - \bar{y}_0)/2$            (61)
    c -   $\delta_1 = h_1(\delta_1 y_1)$                  (62)
    d -   Terminate if $\delta_1 = 0$; otherwise find $y_1$.
    e -   $z_1 = y_0 - y_1M$                              (63)
2 -   Loop for $i = 1,\cdots,n-1$
    a -   $h_{i+1} = h_iM' + \Sigma_{k=1}^{i}\gamma_{ki}h_k$   (64)
    b -   $\chi_i y_{i+1} = z_i - \delta_i y_i$           (65)
    c -   $\chi_i = h_{i+1}(\chi_i y_{i+1})$              (66)
    d -   $z_{i+1} = y_i - y_{i+1}M$                      (67)
    e -   $\delta_{i+1} = h_{i+1}(z_{i+1})$               (68)
```

Theorem 6: Algorithms 4 and 6 are equivalent.

The proof of the theorem follows from the lemma and the definition of M. The advantages of Algorithms 5 and 6 over the preceding ones will be apparent in the following section where matrices will be associated with M and coordinate vectors with the vectors and functionals with the result that varied representations may be selected for polynomials in P_n.

Representation Theory

Let X be a basis for P_n and F be the dual basis in P_n'. The coordinate vectors of a polynomial p_i and a functional f_i will be the row vectors

$$\underline{p}_i = \left[p_{i1}\ \ p_{i2} \cdots p_{in}\right] \tag{69}$$

and

$$\underline{h}_i = \left[h_{i1}\ \ h_{i2} \cdots h_{in}\right] \tag{70}$$

where the p_{ij} and h_{ij} are the coordinates with respect to X and F, respectively. Corresponding to these bases are matrices $\underline{M}$ and $\underline{M}'$ defined by

$$\underline{M} = [M]_X \tag{71}$$

and

$$\underline{M}' = [M']_F. \tag{72}$$

When $m_0(\lambda) = \lambda^n$ and X is the natural basis, the resulting matrix, $\underline{M}$, will be denoted $\underline{J}$. $\underline{J}$ has zero entries except for a subdiagonal of plus 1's. More generally, for arbitrary $m_0(\lambda)$, monic of degree n, and X the natural basis, the resulting $\underline{M}$ is the companion matrix, denoted $\underline{C}$, where

$$\underline{C} = \underline{J} - \underline{e}_1'\underline{m}_0 \tag{73}$$

and $\underline{m}_0 = \begin{bmatrix} m_{01} & m_{02} & \cdots & m_{0n} \end{bmatrix}$. The matrix $\underline{M}$ for any specified basis X can be obtained in a similar manner.

Equations (6), (7), (8), (10), and (12) can be verified by first choosing $\underline{M} = \underline{C}$ and $\underline{R}$, $\underline{S}$, and $\underline{Y}$ unit upper triangular. Then substitution of $\underline{C} = \underline{XMX}^{-1}$ gives them for the general case of a cyclic $\underline{M}$.

It is useful to be able to reverse the procedure so that a basis, say Y, corresponding to a given matrix $\underline{M}$ may be determined such that $\underline{M}_Y = \underline{M}$. Assuming $\underline{M}$ is cyclic, it is known that a matrix $\underline{Y}$ exists such that $\underline{YCY}^{-1} = \underline{M}$. The first column of $\underline{Y}$ is $\underline{f}_1'$ where $\underline{f}_1 = [f_1]_G$, the coordinate vector of the first natural functional f_1 with respect to the basis G which is dual to Y. The formula for the y_i is

$$y_i = \underline{e}_i(\lambda\underline{I} - \underline{M})^a\underline{f}_1', \tag{74}$$

where superscript "a" means adjoint.

To establish Equation (14), observe that $\underline{YCY}^{-1} = \underline{M}$ implies

$$\underline{Y}(\lambda\underline{I}-\underline{C})^a = (\lambda\underline{I}-\underline{M})^a\underline{Y}. \tag{75}$$

Premultiplying both members by $\underline{e}_i$ and postmultiplying by $\underline{e}_1$ gives Equation (74).

Tridiagonal $\underline{M}$ Two particular choices for $\underline{M}$ have been investigated, $\underline{M}$ tridiagonal and $\underline{M}$ diagonal. The first leads to a wide variety of representations. Consider then

$$\underline{M} = tdiag(-\eta_1\ -\theta_1\ ,\ 1\ -\eta_2\ -\theta_2\ ,\ \cdots,1\ -\eta_n). \tag{76}$$

For this choice of $\underline{M}$, $\underline{f}_1$ can be $\underline{e}_1$. Equation (74) then can be employed to identify the y_i in the following cases. When η_i and θ_i are all $= 0$, $\underline{M} = \underline{J}$, and the natural representation is obtained. When $\eta_i \neq 0$ and all $\theta_i = 0$, the y_i are the Newton interpolation polynomials. The η_i need not be distinct. In the special case where all η_i are equal the Taylor's series representation is obtained. The case where all $\eta_i = 0$ and θ_i are real is of special interest in stability analysis and in network synthesis when Equation (13) is satisfied. A relevant theorem will be stated.

Theorem 7: The coordinate vectors of the y_i formed in Algorithm 6, for $i = 1,\cdots,n$, have every other entry equal to zero for the representation defined by $\underline{M}$ tridiagonal and all $\eta_i = 0$, when Equation (13) is satisfied.

The theorem is well known when $\underline{M} = \underline{J}$ because the y_i are alternately even and odd, and it may be established by observing that the coordinates with respect to the more general representation where $\theta_i \neq 0$ will also be alternately zero for even and odd polynomials. The theorem can be generalized to the case where $\underline{M}$ is unit upper Hessenberg and the main, second upper diagonal, fourth upper diagonal, $\cdots$, are zero. A corollary to this theorem is also well known for $\underline{M} = \underline{J}$ and is useful in stability testing because the computations of Equation (7) can be avoided in certain cases.

Corollary The entries $y_{0j} = (-1)^j y_{0j}$, when the hypotheses of Theorem 7 are satisfied.

Algorithm 6 will now be stated for the matrix $\underline{M}$ of Equation (76). The functionals, h_i may be chosen to be the dual basis to Y, so that their coordinate vectors are $\underline{e}_i$. The tests for δ_1 and $\chi_i = 0$ are omitted here.

Algorithm 7: Given $\underline{y}_0$ and $\underline{y}_0^-$ and $\underline{M}$ of Equation (76), find $\underline{T}$.

```
1 - Initialization of loop
    a - $\delta_1 = (y_{01} - y_{01}^-)/2$    (77)
    b - $\underline{y}_1 = (\underline{y}_0 - \underline{y}_0^-)/2\delta_1$    (78)
    c - $y_{n-1,n+1} = y_{n,n+1} \equiv 0$    (79)
2 - Loop for $i = 1,\cdots,n-1$
    a - $\chi_i = y_{i-1,i+1} - \delta_i y_{i,i+1} - \underline{y}_i \underline{M}\underline{e}_{i+1}'$    (80)
          $= y_{i,i+1}(\eta_{i+1} - \delta_i) + y_{i-1,i+1} - y_{i,i+2} + \theta_i$    (81)
    b - $\underline{y}_{i+1} = (\underline{y}_{i-1} - \underline{y}_i(\underline{M} + \delta_i \underline{I}))/\chi_i$    (82)
    c - $\delta_{i+1} = y_{i,i+1} - \underline{y}_{i+1}\underline{M}\underline{e}_{i+1}'$    (83)
          $= y_{i,i+1} - y_{i+1,i+2} + \eta_{i+1}$    (84)
```

Algorithm 2 is seen to be the special case of Algorithm 7 with all η_i and $\theta_i = 0$. To get $\underline{y}_0$ and $\underline{y}_0^-$ from Algorithm 5, let $\underline{r}_n = \underline{e}_n$.

Diagonal M The second choice of $\underline{M}$ to be discussed is for $\underline{M}$ to be a diagonal matrix Λ, where

$$\underline{\Lambda} = diag(\lambda_1, \lambda_2, \cdots, \lambda_n). \tag{85}$$

In this case the λ_i must be distinct or $\underline{M}$ would not be cyclic as hypothesized. If the vector f_1 has coordinates

$$f_{1j} = 1/\pi_{i=1}^{n}(\lambda_j - \lambda_i), \quad omitting\ i = j, \tag{86}$$

the y_i given by Equation (74) are the Lagrange interpolation polynomials. The coordinate vector for $r_n = 1$ is therefore $\underline{r}_n = [1 \;\; 1 \cdots \;\; 1]$. With this choice of $\underline{f}_1$ and $\underline{r}_n$, Algorithm5,6 can be executed for the case where $\underline{M} = \underline{\Lambda}$. A simple choice for the γ_{ki} of Equation (59) is $\gamma_{ki} = 0$ for all k.

Perturbation Theory II

A particular application of Algorithm 7 is in the algorithm for improving upon an approximate solution $\underline{T}_0$ of Equation 1, as discussed in the previous section on perturbation theory. Subject to those assumptions, $\underline{r}_0 = \underline{y}_0$ approximates the zero vector and $\underline{s}_0 = \underline{y}_0^-$ is approximately $2\delta_1$ times $\underline{e}_1$. It can then be seen that all the $\underline{y}_i$ are approximately $\underline{e}_i$, and that cancellation errors will be of second order. This result will be stated as a theorem.

Theorem 8: When $\underline{M}$ in Algorithm 7 is approximately equal to $\underline{T}$, the cancellation errors in this algorithm will be of second order.

It is interesting to observe that when $\underline{M} = \underline{A}$, there will be no cancellation errors in determining $\underline{r}_0$ because $\underline{r}_0 \equiv 0$, but there will be cancellation errors in Algorithm 7. This is just the converse of the situation that obtains when the improvement algorithm is employed where $\underline{M}$ is approximately equal to $\underline{T}$. It was observed in all the experiments, one of which is described in the next section, that $\underline{T}_0$ determined with $\underline{M} = \underline{A}$ could be improved upon by the iterative procedure of this section.

EXPERIMENTS

An experiment will now be described which yielded the five sets of χ_i shown in Table 1. They correspond to the five choices of $\underline{M} = \underline{A}, \underline{T}_0, \cdots, \underline{T}_4$, shown in that order, and therefore illustrate the application of the improvement algorithm. The matrix $\underline{T}_0$ is the solution obtained with $\underline{M} = \underline{A}$. The matrix $\underline{A}$ corresponds to a Bessell polynomial ladder network of order 10 having $\rho_1 = \rho_{10}$ [11;Table 13.8, p. 619]. The calculations were performed on an IBM 360 computer. The program was written in Fortran IV, level G, and computations were performed with 7 decimal digits.

The matrix, $\underline{A}$, is the tridiagonal matrix with nonzero entries

$$\alpha_{11} = 1.586000, \quad \alpha_{10,10} = 53.47600,$$

```
I    CHI1(I)      CHI2(I)      CHI3(I)      CHI4(I)      CHI5(I)

1  960.0693359  960.0690918  960.0686035  960.0686035  960.0686035

2  197.0594177  197.0698090  197.0702057  197.0700836  197.0701599

3   88.5478668   88.5239868   88.5239563   88.5240784   88.5240326

4   52.3874512   52.4009247   52.4008636   52.4009247   52.4009247

5   35.9551086   35.9389496   35.9389954   35.9389496   35.9389648

6   27.1763763   26.7710114   26.7710266   26.7710114   26.7709808

7   19.6021118   20.3030090   20.3030090   20.3029938   20.3029938

8   16.1674042   13.9730158   13.9730043   13.9730110   13.9730101

9    4.3439941    5.2832985    5.2833014    5.2832994    5.2833004
```

Table 1

and, for $\alpha_{i,i+1}$, in the order $i = 1, \cdots, 9$

5.283300, 13.97300, 20.30300, 26.77100, 35.93900,
52.40100, 88.52400, 197.0700, 960.0700.

The entries in $\underline{B}$ are the same except for one sign change so that $\beta_{10,10} = -53.47600$.

The characteristic function of a tridiagonal matrix is not changed if the order of the entries within its diagonals is reversed, so it is seen that $\underline{A}$ and $\underline{B}$ specify $\underline{T}$ and $\underline{T}^{-}$ having the same entries as $\underline{A}$ and $\underline{B}$ but in reversed order. Examination of Table 1 shows a considerable loss of accuracy in the first column where $\underline{M} = \underline{A}$, but the succeeding columns in all cases duplicate the first 5 digits and, in some cases, the sixth. The entries α_{11} and $\alpha_{10,10}$, which aren't shown, were duplicated to six and five digits respectively. The improvement in the values of the χ_i appears to have been realized in the first step of the improvement algorithm, but the values of $\underline{r}_0$ determined in each of the following steps were observed to be increasingly close to zero, indicating that some further improvement exists in the succeeding steps.

Experiments which yielded $\underline{T}$ satisfying Equation (13) were performed.* The method of Reference [6], which is based on the H.R.Schwarz' algorithm was employed to obtain a $\underline{T}$ satisfying Equation (13) and the Butterworth eigenvalues, which lie on the unit circle in the left half of the λ plane. For n=15, 13 χ_i had 5 and 6 digit accuracy and one had 4. The methods of this paper, including the use of the improvement algorithm, yielded an extra digit accuracy in 7 instances. Another choice of $\underline{M}$, which was nearly as effective as when $\underline{M}$ was an approximation to $\underline{T}$, was the choice obtained from $\underline{A}$ (which was tridiagonal of the form shown in Equation (20)) by replacing all the main diagonal entries with zeros. Theorem 7 was then applicable.

A variant upon the preceding experiment is to reverse the order of the entries in the diagonals of $\underline{B}$ at the outset so that during the improvement procedure the determination of $\underline{s}_0$ approaches an identity and cancellation errors won't be introduced here. This isolates the cancellation errors that occur in determining $\underline{r}_0$. Similarly, the entries in $\underline{A}$ can be reversed to isolate the cancellation occuring in determining $\underline{s}_0$.

CONCLUSION

The tridiagonalization procedure that is presented in this paper is a generalization of currently employed algorithms for finding the continued fraction determined by a reflection coefficient and algorithms for tridiagonalizing a matrix.† The development of the procedure for a polynomial space was seen to unify the two approaches and to facilitate the choice of different representations for the polynomials. The use of representations defined by approximations to the desired tridiagonal matrix was the basis for an algorithm for improving upon the entries in a tridiagonal matrix.

The classical method of ladder network synthesis and the state-variable approach (similarity transformation of a matrix in normal form to tridiagonal form) can be obtained as special cases of the synthesis procedure that is described. They are seen to be closely related procedures, which differ primarily in the representation of the polynomials which are formed.

ACKNOWLEDGMENT

The author wishes to thank Professor E.F.Bolinder for the opportunity to do this work at Chalmers University of Technology, to T.Brodén, S.Petersson, and S.Sigurdh for their programming assistance, and to Mrs W.G.Sauter for typing the manuscript.

REFERENCES

1 - A.S.Householder, *The Theory of Matrices in Numerical Analysis.* New York: Blaidell, 1964.

2 - J.H.Wilkinson, *The Algebraic Eigenvalue Problem.* London: Oxford University Press, 1965.

3 - H.R.Schwarz, "Ein Verfahren zur Stabilitätsfrage bei Matrizen-Eigenwert-Problemen," *Z.angew.Math.u.Phys.*, vol.7, pp 473-500, 1956.

4 - T.G.Marshall,Jr., "Discussion," *Proc.Allerton Conf.*, pp.175-200, Sept.1964.

5 - T.G.Marshall,Jr., "Ladder network synthesis by similarity transformation," *Proc.Allerton Conference*, p. 200, October 1965.

6 - T.G.Marshall,Jr., "Primitive matrices for doubly terminated ladder networks," *Proc.Allerton Conference*, pp. 935-943, October 1966.

7 - I.Navot, "The synthesis of certain subclasses of tridiagonal matrices with prescribed eigenvalues," *SIAM J.on Appl.Math.*, pp. 241-251, Mar 1967.

8 - T.Muir, *A Treatise on the Theory of Determinants* (revised edition by W.H.Metzler). New York: Dover, 1960.

9 - G.Birkhoff and S.MacLane, *A Survey of Modern Algebra* (3rd edition). New York: Macmillan, 1965.

10 - R.M.Thrall and L.Tornheim, *Vector Spaces and Matrices.* New York: Wiley, 1957.

11 - L.Weinberg, *Network Analysis and Synthesis.* New York: McGraw-Hill, 1962.

12 - G.Szentirmai, "Theoretical basis of a digital computer program for filter synthesis," *Proc. Allerton Conference on Circuit and System Theory*, pp. 37-49, November 1963.

13 - J.A.C.Bingham, "A new method of solving the accuracy problem in filter design," *IEEE Trans. on Circuit Theory*, pp. 327-341, September 1964.

---

* An alternate approach to the accuracy problem, which is used in filter design is well known |12,13| and has application to the ladder synthesis problem described herein when $\delta_i = 0$ for $i = 2,\cdots,n-1$.

† The Lanczos' algorithm can be obtained from the choice $\underline{M} = \underline{A}$, and $\gamma_{ii} = \delta_i$, $\gamma_{i-1,i} = \chi_{i-1}$, and $\gamma_{ki} = 0$ for $k < i-1$ in Equation (59).

COMPUTER DESIGN OF MULTISTAGE TRANSISTOR BIAS CIRCUITS

M. A. MURRAY-LASSO and W. D. BAKER*
Bell Telephone Laboratories, Incorporated
Whippany, New Jersey

ABSTRACT

This paper presents some methods and describes a program (BIAS) for automatically calculating on a digital computer the values of the resistors for biasing transistor circuits. As a typical case, a direct coupled multistage transistor amplifier to be built with hybrid integrated circuits (silicon chip transistors and tantalum thin film resistors) is designed.

INTRODUCTION

In general, transistor manufacturers provide a-c data on their transistors only for specified d-c operating conditions. In the case of integrated circuits, in which last minute trimming is not possible, the designer has to be able to predict the a-c behavior of an amplifier accurately and is thus practically forced to use the operating conditions for which he has data. For this reason, in integrated circuit design, biasing should be done with considerable care.

At the present moment, it is difficult to build large capacitors with thin films; hence, it is desirable to avoid them as much as possible. For this reason, there is considerable interest in designing direct coupled multistage transistor amplifiers. The determination of the biasing resistors in a direct coupled multistage transistor amplifier is not quite as simple as it is for a capacitor coupled amplifier. This is due to the fact that the various feedback loops formed yield complicated sets of simultaneous equations which have to be solved. The possibility of programming a digital computer to do automatically the calculations involved becomes attractive. If, besides doing the analysis on the computer, an optimization program is coupled to the analysis routine, it is possible to do the design of the bias circuit automatically. The program may include the possibility of minimizing different performance criteria such as: sensitivity, Mandex worst case conditions, or statistical sensitivity.

By properly formulating the problem, it is possible to attack the biasing of transistor circuits with enough generality to include a wide variety of applications.

EXPLANATION OF THE METHOD USED

The problem of biasing a transistor circuit is a problem in active synthesis. If each transistor is replaced by an equivalent d-c circuit containing current controlled current sources and resistors, then the biasing problem consists of finding a resistive network which, when connected to the active devices plus some independent voltage sources, will produce specified currents in certain branches and specified voltage differences between certain nodes. Generally, because of a-c considerations, additional constraints, such as maintaining certain resistor ratios constant, are imposed. The solution to this synthesis problem, when it exists, is generally not unique; thus taking advantage of this freedom, the designer would like to minimize some additional criterion such as sensitivity.

---

* Presently with the Electrical Engineering Department, University of West Virginia, Morgantown, West Virginia.

For capacitor coupled transistor amplifiers, each transistor is biased independently and it is generally possible to do most of the work analytically since the circuits involved are simple. When the capacitors are taken out, however, the whole network has to be solved simultaneously, specially if feedback between the last and first stage is used. The complexity of the problem increases so much that one rapidly loses the hope of obtaining analytical solutions.

Having settled for a numerical method, it is convenient to use a formulation which is general enough to encompass many applications. This can be done by an iterative scheme using a general d-c circuit analysis routine and an optimization program both with enough flexibility to allow differences in circuit structures, design criteria and constraints.

The biasing problem may be solved in the following way: Assume a network structure and a set of values for the resistors as the first trial of a biasing network using the best information available. With the help of an analysis program, calculate the currents and voltages of the first trial circuit. Of particular interest are the collector current and collector-emitter voltage of each transistor. In general, the analysis will reveal that the operating points are not the desired ones. Using a positive monotonic function of the differences between the desired operating points and the actual values as a criterion of performance, vary the available resistors within their allowable ranges according to some strategy until the analysis routine indicates that the transistors are properly biased. Hopefully, if there are enough variable resistors which can be varied through a sufficiently wide range, it will be possible to achieve the operating points exactly.

Strictly speaking, biasing is a nonlinear circuit problem since the transistors have nonlinear characteristics. In order to obtain "true" intermediate solutions while adjusting the resistors, a nonlinear d-c analysis program and a nonlinear d-c model of the transistor are necessary. This makes the method expensive in computation time since each resistor configuration requires several iterations to obtain a solution. The designer, however, is not interested in the intermediate solutions except in as much as they provide a road to the final resistor biasing network. Therefore, there is no need of obtaining "true" solutions for the intermediate steps as long as the last step does provide a correct solution. This taken into consideration, it is possible to bias the circuit using a linear d-c analysis program using considerably less computation.

The model to be used for the transistor is a small signal d-c linear model around the desired operating point. Although all the intermediate calculations leading to the final biasing network are incorrect because the model of the transistor is only good for the desired operating point, as long as the desired operating point is ultimately achieved, the final calculation will be correct and the final set of resistors will provide a valid biasing circuit.

ANALYSIS PROGRAM

An analysis program which is going to be used many times during a design should be quite efficient. Computationally, the most expensive operations in a general d-c circuit analysis program are matrix multiplication and matrix inversion. In the analysis program, multiplication of topological matrices and parameter matrices is avoided by adding the nonzero portions of the indefinite admittance matrices of each resistor and each controlled current source and by converting each independent voltage source into a Norton equivalent independent current source. For the matrix inversion, a Gauss' type elimination algorithm is used. In the interest of speed, the largest pivotal element in the matrix is not searched. In spite of this, no serious numerical accuracy problems have been encountered up to the present moment with the circuits that have been designed. The analysis routine is able to calculate all the node voltages and branch currents in a circuit of the complexity of that shown in Fig. 5 fifty times per second on a GE-635 computer. A modified version of the analysis program has been implemented on the GE and IBM teletypewriter time shared systems for on-line optimization of d-c circuits.[2] In this version, the user can change the circuit topology on line.

OPTIMIZATION ROUTINE

Any optimization routine must contain three distinct components: an optimization strategy, an error expression to be minimized, and an analysis routine for evaluating the error expression. The optimization strategy observes the error function, decides how to reduce the error, and then takes the steps necessary to carry out the reduction without violating any problem constraints. There are many different optimization strategies, which are iterative in nature.[3] The error expression must be carefully chosen for each problem to assure meaningful results.

Direct pattern search[1] was chosen as the optimization strategy. This particular strategy is easily programmed and its performance is relatively undisturbed by the presence of sharp irregularities in the parameter space. Other strategies may be more useful for certain problems and further investigation of this point is planned. Direct pattern search has two major parts: the exploratory move and the pattern move. Briefly, the strategy may be described as follows. An initial guess is made for the value of each of the parameters of a system and some error expression is evaluated for that initial parameter set. Each parameter in turn is varied slightly about its initial value. The effect of each variation on the error expression is observed, and those changes that reduce the error are retained. This yields a slightly modified set of parameters which defines the "unit vector" of a move in parameter space which will decrease the error. This process is the exploratory move. The next step is to make a larger change in the parameter value in the direction indicated by the exploratory move. This is a pattern move. If the pattern move succeeds in reducing the error, a second pattern move is made in the same direction. Pattern moves continue until one fails to reduce the error. At that point, a new exploratory move is made to find a new direction for improvement. The optimization terminates when an exploratory move is unable to further reduce the error. A simplified flow chart for pattern search is shown in Fig. 1 and one for the exploratory move in Fig. 2. A simple pattern search is illustrated in two dimensions in Fig. 3.

DESIGN CRITERIA

A program for the design of transistor bias networks must be able to consider certain design criteria. The program should be able to calculate the resistors necessary to achieve the desired operating points. It should be possible to place constraints on the allowable resistance values in order to insure useable solutions. This is especially important for integrated or thin-film circuits where resistance values must be compatible with fabrication techniques. The error expression used in the optimization should allow the emphasis of certain terms over the others. This weighting of terms allows the most important bias current, for example, to be given the most influence on the optimization process. The program should also be able to consider sensitivity or worst-case analysis in the selection of resistance values.

The direct pattern search program described in this paper can satisfy all of these criteria. The bias resistances can be calculated subject to constraints, the error terms can be given relative weights, and sensitivity or worst-case calculations can be included in the optimization.

This program has been applied to the design of thin-film, very wideband transistor feedback amplifiers. The acceptability of constraints by the d-c optimization program allows those resistances which are critical to the a-c performance to be fixed, or held within rigid limits, during the d-c design. Of considerable interest in such amplifiers is the variation of the operating point with changes in transistor parameters, especially h_{FE} (or "beta"). A detailed investigation of the variation of operating points in feedback amplifiers with respect to betas was conducted so that this information might be included in the optimization. The investigation revealed a highly wavy "beta space." The result is that the partial derivatives of the currents or voltages at any point can be highly misleading as to the behavior at a slightly different point. Since the partial derivatives are meaningless under these conditions, sensitivity analysis and the Mandex worst-case analysis fail. A reasonably typical plot of the "beta space" for a three transistor amplifier is shown in Fig. 4. Here the collector current of the last stage is plotted as a function of the betas of the first two stages with the third stage beta fixed at a fairly typical value. As an alternative approach to obtaining some worst-case data to include in the optimization, a Monte Carlo worst-case analysis was adopted. The transistor betas were chosen randomly (from a uniform distribution for lack of better information) and the resulting sets of currents and voltages were searched to find the worst-case conditions. This or similar approaches appear to be necessary to obtain useful worst-case information for an apparently large class of circuits for which the usual methods fail.

PROGRAM DESCRIPTION

The optimization program (BIAS) for the design of transistor bias networks was written in Fortran IV for the GE-625/635 computing system. The input to the program consists of the equivalent circuit connection topology, source information, initial estimates for each network resistance, the maximum and minimum allowable values for each resistance, and the d-c operating point desired for each transistor. In addition, other error criteria such as worst-case collector current limits can be specified for a particular problem. The program output is a listing of the final computed resistances for the network, the node-to-ground voltages, the element currents, the actual operating point for each transistor, and any worst-case information. The analysis routine is a DC linear analysis program. It will accept resistances, current-controlled current sources, and independent voltage sources. Any transistor model which is composed of these elements is acceptable to the program. Currently, the program will handle networks with a maximum of 30 branches, 15 nodes, and 10 transistors.

This capacity is set by memory space allocation (and ultimately by computation time).

It is not possible to predict the successful convergence of a pattern search optimization to the true optimum for a general problem. It is usually very difficult even to detect when a true optimum has been achieved. For these reasons, it is not possible to guarantee that all possible bias circuits can be properly optimized. The program has, however, been quite successful for the three or four stage circuits investigated. Usually, the actual and desired bias points are essentially identical. Occasionally, the desired operating points may not be physically realizable because of circuit constraints. Such circuits have been optimized to a better solution than any obtained by hand calculation.

A typical transistor amplifier is shown in Fig. 5 along with its equivalent circuit. The program was arranged to calculate the network resistances (subject to certain constraints) necessary to achieve specified transistor operating points. In particular, I_C = 10 mA and V_{CE} = 5 v were specified. In this particular case, the program was able to achieve exactly the desired operating points. A copy of the program output is shown in Fig. 6.

A number of two to four stage, direct-coupled feedback amplifiers have been considered and appropriate bias networks have been successfully designed. Some of these designs have also included the minimization of the worst-case collector currents with variations in the transistor forward current transfer ratio.

CONCLUSIONS

A direct pattern search optimization routine and an efficient d-c linear analysis routine have been written and coupled. The resultant program (BIAS) is capable of calculating values for the bias resistors necessary to achieve specified operating points for the transistors in direct-coupled, multistage circuits. Upper and lower constraints can be applied to any of the resistance values. The program has consistently converged to the solution for a number of two and three stage circuits. The feasibility of using optimization methods to design active networks has been demonstrated. Similar programs using a-c analysis routines should permit a-c design optimization.

The well-known Mandex worst-case analysis has been shown to fail for a class of feedback amplifiers. This suggests that other methods, such as statistical ones may be required for any worst-case or sensitivity analysis.

REFERENCES

1. Hook, R., and T. A. Jeeves, "Direct Search" Solution of Numerical and Statistical Problems, Journal of the Association for Computing Machinery, Vol. 8, No. 2, April 1961, pp. 212-229.

2. Murray-Lasso, M. A., "On-Line Optimization," Symposium on Computer Aids for Large Circuit Arrays, University of Wisconsin, Madison, Wisconsin, September 1967.

3. Fleischer, P. E., "Optimization Techniques in System Design," Chapter 6 in System Analysis by Digital Computer, edited by F. F. Kuo and J. F. Kaiser, Wiley, New York, 1966.

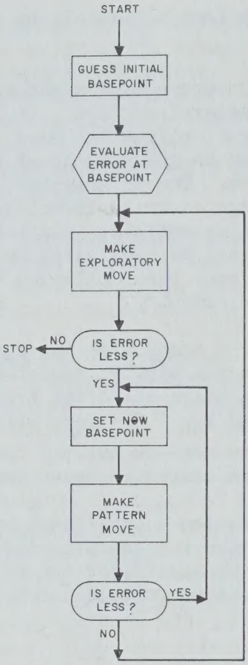

START
GUESS INITIAL BASEPOINT
EVALUATE ERROR AT BASEPOINT
MAKE EXPLORATORY MOVE
IS ERROR LESS ?
NO
STOP
YES
SET NEW BASEPOINT
MAKE PATTERN MOVE
IS ERROR LESS ?
YES
NO

FIGURE I PATTERN SEARCH

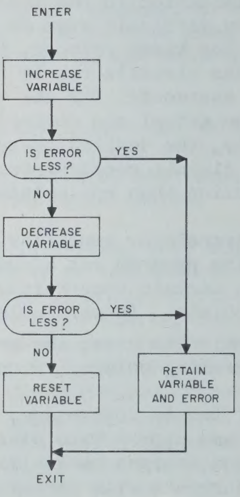

ENTER
INCREASE VARIABLE
IS ERROR LESS ?
YES
NO
DECREASE VARIABLE
IS ERROR LESS ?
YES
NO
RESET VARIABLE
RETAIN VARIABLE AND ERROR
EXIT

FIGURE 2 EXPLORATORY MOVE

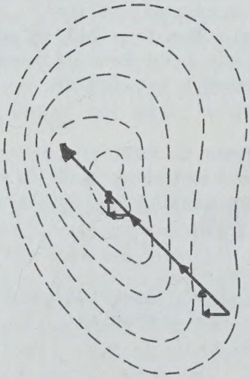


FIGURE 3
A SIMPLE PATTERN SEARCH

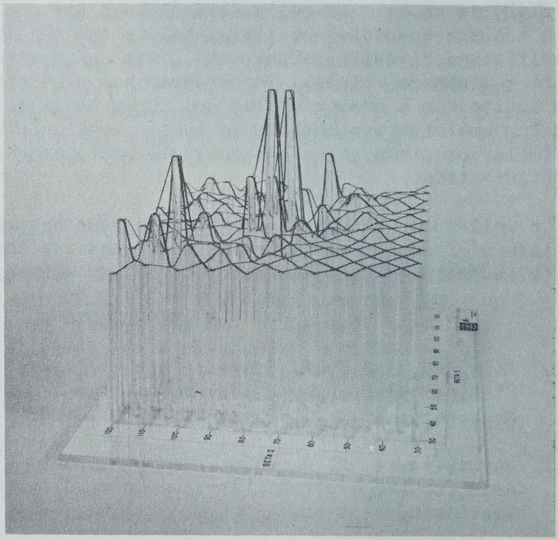

BETA 2
BETA 1

FIGURE 4 ASPECT OF THE β TERRAIN

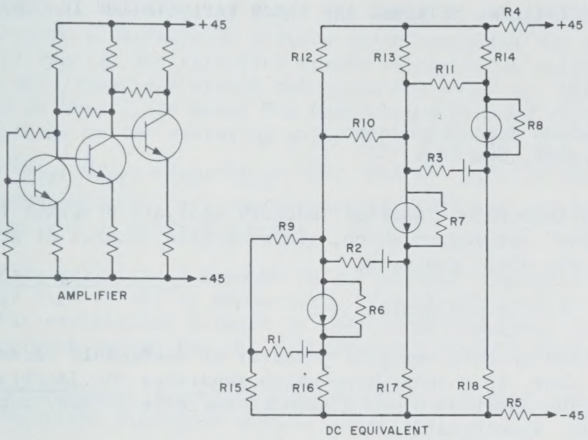


FIGURE 5 - TYPICAL PROBLEM

```
    INITIAL BRANCH RESISTANCES
  1-5    0.2000E 03    0.2000E 03    0.2000E 03    0.1000E 01    0.1000E 01
  6-10   0.1000E 05    0.1000E 05    0.1000E 05    0.5000E 04    0.5000E 04
 11-15   0.5000E 04    0.3000E 04    0.3000E 04    0.3000E 04    0.5000E 04
 16-18   0.3000E 04    0.3000E 04    0.3000E 04

    EXPLORATORY MOVES=  48         PATTERN MOVES=   105

       NODE VOLTAGES                                  ELEMENT CURRENTS

    NODE          VOLTAGE                        BRANCH            CURRENT
     1           -0.9392E 01                        1             0.3169E-03
     2           -0.1016E 02                        2             0.3169E-03
     3           -0.5151E 01                        3             0.3166E-03
     4           -0.5914E 01                        4             0.3488E-01
     5           -0.9149E 00                        5             0.3488E-01
     6           -0.1678E 01                        6             0.5004E-03
     7            0.3321E 01                        7             0.4999E-03
     8            0.4497E 02                        8             0.5000E-03
     9           -0.4497E 02                        9             0.4241E-02
                                                   10             0.1136E-02
                                                   11             0.9761E-03
                                                   12             0.1343E-01
                                                   13             0.1048E-01
                                                   14             0.1097E-01
                                                   15             0.3924E-02
                                                   16             0.1032E-01
                                                   17             0.1032E-01
                                                   18             0.1031E-01

      FINAL BRANCH RESISTANCES
    1-5    0.2000E 03    0.2000E 03    0.2000E 03    0.1000E 01    0.1000E 01
    6-10   0.1000E 05    0.1000E 05    0.1000E 05    0.1000E 04    0.3730E 04
   11-15   0.4340E 04    0.3732E 04    0.4377E 04    0.3795E 04    0.9065E 04
   16-18   0.3372E 04    0.3783E 04    0.4197E 04

       OPERATING POINTS

    VCE (VOLTS)
       0.500E 01    0.500E 01    0.500E 01

    IC (MA)
       0.100E 02    0.100E 02    0.100E 02
```

FIGURE 6

TWO CLASSES OF VARIATIONAL NETWORKS AND THEIR EXPLOITATION IN COMPUTER AIDED DESIGN

G. D. Hachtel
IBM
Thomas J. Watson Research Center
Yorktown Heights, New York

An essential feature of "package" network analysis programs is an "equation generator" subprogram which, given certain literal or graphical input, generates the state equations

$$\dot{\underset{\sim}{x}} = \underset{\sim}{f}(\underset{\sim}{x}) \;, \qquad (1)$$

where x is the state vector, and $\underset{\sim}{p}$ is a vector of designable parameters. Not included, to date, is a subprogram which generates the Jacobian matrix, $\partial f/\partial x$, as well. The Jacobian plays a significant role in many aspects of numerical analysis, specifically,

a) variational calculus for sensitivity and optimization [1], and

b) Newton-like iteration in algorithms for fast integration of differential equations with both large and small time constants [2].

Presented below is a discussion of two classes of variational (linear) networks, which are derived from, and have the same topology as, a parent (nonlinear) network. The A-matrix of networks in the first variational class is the Jacobian of the parent network; the A-matrix of circuits in the second class is the transpose of the Jacobian, $(\partial \underset{\sim}{f}/\partial \underset{\sim}{x})^T$.

The use of variational networks enables the programmer to utilize his equation generator, or his whole analysis package, to avoid the labor of differentiating $\underset{\sim}{f}(x)$ with respect to x. With the variational networks, the required differentiation is done at the input-element level. In a sense, this means that the sparseness of the network equations prior to being put into the normal form (1) is being exploited. A complete description of a 5-element kind (R,L,C,E,J) network is illustrated in Figs. 1 and 2.

Figure 1 illustrates the topological constraints. The matrix F is the fundamental voltage loop equation in the circuit, so that Fig. 1 represents Kirchhoff's current (KCL) and voltage (KVL) laws. Figure 2 illustrates the branch-constitutive relations which establish relations between the branch terminal variables. For an RC network the equations of Figs. 1 and 2 reduce to the state equations

$$\dot{\underset{\sim}{V}}_C = -\mathcal{C}^{-1} F_{RC}^T \mathcal{R}^{-1} F_{RC} \underset{\sim}{V}_C \;, \qquad (2)$$

where, using the notation of Kuh and Rohrer [3], $\mathcal{C}$ and $\mathcal{R}$ are symmetric element matrices, and F_{RC} is the link-resistance, tree-capacitance partition of F. Equation (2) illustrates that f is always the product of several topological and circuit element matrices. Thus even for a rather sparse system in which $\mathcal{C}$ and $\mathcal{R}$ are sparse, $\partial \underset{\sim}{x}$ is in general a full matrix and hence very messy to differentiate.

Given $\underset{\sim}{x}$, the Jacobian $\partial \underset{\sim}{f}/\partial \underset{\sim}{x}$ can be obtained (with the help of the programs, state equation generation) without directly differentiating messy expressions like that of (3). The trick is to obtain from the original network a variational network which satisfies the topological relations of Fig. 1, but satisfies branch constitutive relations like that of Fig. 2, but linearized around the operating point $\underset{\sim}{x}$. This process is illustrated in Fig. 3, in which the branch-constitutive relations are expressed in terms of the operating point $\underset{\sim}{V}_{TB}$, $\underset{\sim}{V}_{LK}$, $\underset{\sim}{I}_{TB}$, $\underset{\sim}{I}_{LK}$ and the variations $\delta\underset{\sim}{V}_{TB}$, $\delta\underset{\sim}{V}_{LK}$, $\delta\underset{\sim}{I}_{TB}$, $\delta\underset{\sim}{I}_{LK}$, etc. Without loss of generality in the cases illustrated below, all resistive nonlinearities are treated as dependent current or voltage sources.

The state-variational network is the network described by the equations of Fig. 3, and by those (with $\underset{\sim}{V}$ and $\underset{\sim}{I}$ replaced by $\delta\underset{\sim}{V}$ and $\delta\underset{\sim}{I}$) of Fig. 1. This variational network is seen to be identical to the parent nonlinear network except for the following two details (Fig. 4):

1. The nonlinear dependent sources of Fig. 2 are replaced by linearized dependent sources (Fig. 4a).
2. A linear dependent current generator is added in parallel with each nonlinear capacitor (Dual statements apply to any nonlinear inductance present) (Fig. 4b).

The A-matrix of the state-variational network is the desired Jacobian. This can be shown by eliminating all of the 16 kinds of branch variables from the equations of Figs. 1 and 3 except the chosen state variables $\delta\underset{\sim}{V}_C$ and $\delta\underset{\sim}{I}_L$. At equilibrium ($\dot{\underset{\sim}{x}} = f(\underset{\sim}{x}) = 0$) the reactive branch expressions simplify, for example, for the kth tree branch capacitance

$$\delta I_{CK}\Big|_{\dot{V}_{C=0}} = C_{CK}\delta\dot{V}_{CK} + \frac{\partial C_{CK}\dot{V}_{CK}}{\partial \underset{\sim}{V}_C}\,\delta V_C\Big|_{\dot{V}_C=0} = C_C\delta\dot{\underset{\sim}{V}}_C\,, \qquad (3)$$

where it should be noted that the second term of (3) is an inner product. In this case the eigenvalues of the Jacobian are the natural frequencies of the linearized network. Away from equilibrium ($\dot{x} \neq f(\underset{\sim}{x})$), the eigenvalues of the Jacobian at a given time control the asymptotic stability of the numerical integration, and plays a key role in many other aspects of numerical analysis.

A second class of variational network arises in problems employing the calculus of variations to minimize a performance functional of the form

$$P(\underset{\sim}{p}) = \int_0^{t_f} \mathcal{L}\,dt = \int_0^{t_f} \{h(\underset{\sim}{x}[\underset{\sim}{p},t],\underset{\sim}{p},t) + \underset{\sim}{\lambda}^T(\dot{\underset{\sim}{x}}-\underset{\sim}{f})\}dt\ , \qquad (4)$$

where $\underset{\sim}{p}$ is a vector of time-invariant design parameters, and $\underset{\sim}{\lambda}^T$ is a vector of Lagrange multipliers. Under the constraint $\dot{\underset{\sim}{x}}-\underset{\sim}{f} = 0$, $P = \int_0^{t_f} h\,dt$, so h is seen to be an instantaneous performance function. It has been shown that the gradient of P in $\underset{\sim}{p}$-space can be found as shown in Fig. 5, subject to possible transversality conditions on the boundary values $\lambda^T(0)$ and $\lambda^T(t_f)$ [1]. Again the design technique requires the implementation of the Jacobian, which can be achieved by introducing the adjoint-variational network. The Lagrange multiplier $\underset{\sim}{\lambda}$ is the state vector of this network, and the Jacobian the transpose of its A-matrix. Once the variational network is found, a package program such as Sceptre can be used to determine, in succession, $\underset{\sim}{x}(t)$, P, $\underset{\sim}{\lambda}(t)$, and $\partial P/\partial \underset{\sim}{p}$. This technique is important in optimization routines which require repeated evaluations of P and $\partial P/\partial \underset{\sim}{p}$.

In the linear case, the adjoint-variational network is just the original network with all resistances replaced by their negatives [4]. In the nonlinear case the modifications are more extensive. The key step is to introduce a Lagrange multiplier for each of the 16 constraint functions ϕ_K in Figs. 1 and 2. The performance function then becomes, instead of (4),

$$P = \int_0^{t_f} (h + \sum_K \lambda_K^T \phi_K) dt \quad . \tag{6}$$

There are 32 Euler-Lagrange equations arising in the minimization of this functional. The first (16) are those of Figs. 1 and 2, namely $\phi_K = 0$, $K = 1,16$. The other 16 are codified by the substitutions of Fig. 6. The variables of the (linear) adjoint network are designated with a hat (^) to distinguish them from the similarly subscripted variables of the parent (nonlinear) network.

Euler equations (17-24 of the 32 derived from (6)) turn out to be

$$\underset{\sim}{\hat{V}}_{LK} + F\underset{\sim}{\hat{V}}_{TB} = 0 \tag{7}$$

$$\underset{\sim}{\hat{I}}_{TB} - F^T \underset{\sim}{\hat{V}}_{LK} = 0 \tag{8}$$

Thus, by comparison to the equations of Fig. 4, the adjoint network has the same topology as the parent network. As before, however, the branch-constitutive relations are modified; in this case, as indicated in Fig. 7. The transformation from parent to adjoint network involves 3 phases:

1. The original current (voltage) sources are opened (shorted) (Fig. 6a).

2. All resistances are replaced by their negatives (Fig. 6b).

3. All nonlinear capacitances are changed to time varying capacitances, and a current source

$$J_{CK} = \frac{\partial h}{\partial V_{CK}} + \underset{\sim}{\hat{V}}_J \frac{\partial \underset{\sim}{I}}{\partial V_{CK}} + \hat{I}_E \frac{\partial \underset{\sim}{E}}{\partial V_{CK}} - \hat{V}_C \frac{\partial C\dot{\underset{\sim}{V}}_C}{\partial V_{CK}} \tag{9}$$

   is added in parallel. Dual statements apply for inductors.

Note that $\hat{V}_J$ is the vector of voltages across the open current sources in the adjoint network (dual statement for $\hat{I}_E$), and that $\hat{V}_J \, \partial \underset{\sim}{I}/\partial V_{CK}$ is an inner product.

An interesting point is that in general

$$\frac{d}{dt} C\hat{V}_C - \hat{V}_C \frac{\partial C\dot{\underset{\sim}{V}}_C}{\partial V_C} \neq C\,\dot{\hat{V}}_C \tag{10}$$

unless, for example, each capacitor depends only on its own voltage. This is surprising in that one would expect only the incremental capacitance C in a variational network.

Using the variational networks described above [12], Sceptre has been used to generate a fortran program for a model 20-branch problem with a 5-dimensional state vector. The program alternately integrates the state and adjoint equations in order to obtain the information needed to obtain P and $\partial P/\partial \underset{\sim}{p}$ (Fig. 5).

REFERENCES

1. G. D. Hachtel and R. A. Rohrer, "Variational Methods for the Optimal Design and Synthesis of Switching Circuits," to be published in the Proceedings of the IEEE, Vol. 55, No. 11, November 1967.

2. R. A. Willoughby, F. G. Gustavson and W. L. Liniger, "An Optimal Crout Algorithm for Solving Sparse Systems of Linear Algebraic Equations," to be published.

3. E. S. Kuh and R. A. Rohrer, "The State Variable Approach to Network Theory," Proceedings of the IEEE, Vol. 53, No. 7, July, 1965.

4. R. A. Rohrer, "Synthesis of Arbitrarily Tapered Lossless Transmission Lines," Symposium on Generalized Networks, Polytechnic Institute of Brooklyn, April, 1966.

$$\underset{\sim}{\phi}_{1-4} = \underset{\sim}{V}_{LK} + F\underset{\sim}{V}_{TB} = 0$$

$$\underset{\sim}{\phi}_{5-8} = \underset{\sim}{I}_{TB} - F^{T}\underset{\sim}{I}_{LK} = 0$$

$$\underset{\sim}{V}_{TB} \triangleq \underset{\sim}{V}_{C}, \underset{\sim}{V}_{G}, \underset{\sim}{V}_{\Gamma}, \underset{\sim}{V}_{E}$$

$$\underset{\sim}{I}_{LK} \triangleq \underset{\sim}{I}_{S}, \underset{\sim}{I}_{R}, \underset{\sim}{I}_{L}, \underset{\sim}{I}_{J}$$

Topological Relations:

Kirchoff's Laws

Fig. 1

$$\begin{aligned}
\phi_{9} &= \underset{\sim}{E}(\underset{\sim}{V}_{C}, \underset{\sim}{I}_{L}, t) - \underset{\sim}{V}_{E} &&= 0\\
\phi_{10} &= \underset{\sim}{I}(\underset{\sim}{V}_{C}, \underset{\sim}{I}_{L}, t) - \underset{\sim}{I}_{J} &&= 0\\
\phi_{11} &= G_{R}\underset{\sim}{V}_{R} - \underset{\sim}{I}_{R} &&= 0\\
\phi_{12} &= R_{G}\underset{\sim}{I}_{G} - \underset{\sim}{V}_{G} &&= 0\\
\phi_{13} &= C_{S}\dot{\underset{\sim}{V}}_{S} - \underset{\sim}{I}_{S} &&= 0\\
\phi_{14} &= C_{C}\dot{\underset{\sim}{V}}_{C} - \underset{\sim}{I}_{C} &&= 0\\
\phi_{15} &= L_{L}\dot{\underset{\sim}{I}}_{L} - \underset{\sim}{V}_{L} &&= 0\\
\phi_{16} &= L_{\Gamma}\dot{\underset{\sim}{I}}_{\Gamma} - \underset{\sim}{V}_{\Gamma} &&= 0
\end{aligned}$$

Branch Constitutive Relations

Fig. 2

$$\frac{\partial \underset{\sim}{E}}{\partial \underset{\sim}{V}_C}\,\delta \underset{\sim}{V}_C + \frac{\partial E}{\partial \underset{\sim}{I}_L}\,\delta I_L - \delta \underset{\sim}{V}_E = 0$$

$$\frac{\partial \underset{\sim}{I}}{\partial \underset{\sim}{V}_C}\,\delta \underset{\sim}{V}_C + \frac{\partial \underset{\sim}{I}}{\partial \underset{\sim}{I}_L}\,\delta \underset{\sim}{I}_L - \delta \underset{\sim}{I}_J = 0$$

$$G_R\,\delta \underset{\sim}{V}_R - \delta \underset{\sim}{I}_R = 0$$

$$R_G\,\delta \underset{\sim}{I}_G - \delta \underset{\sim}{V}_G = 0$$

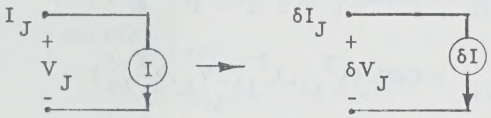
$$C_S\,\delta \dot{\underset{\sim}{V}}_S + \frac{\partial C_S \underset{\sim}{V}_S}{\partial \underset{\sim}{V}_S}\,\delta \underset{\sim}{V}_S - \delta \underset{\sim}{I}_S = 0$$

$$C_C\,\delta \dot{\underset{\sim}{V}}_C + \frac{\partial C_C \dot{\underset{\sim}{V}}_C}{\partial \underset{\sim}{V}_C}\,\delta V_C - \delta \underset{\sim}{I}_C = 0$$

Variational branch constitutive relations.

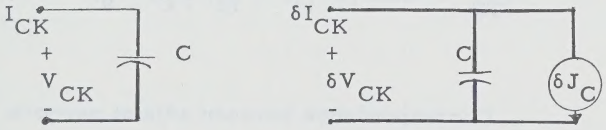
Fig. 3

a)



$$I_J = I(\underset{\sim}{V}_C, \underset{\sim}{I}_L, t) \qquad \delta I = \frac{\partial I}{\partial \underset{\sim}{V}_C}\,\delta \underset{\sim}{V}_C + \frac{\partial I}{\partial \underset{\sim}{I}_L}\,\delta \underset{\sim}{I}_L$$

b)



$$\delta J_C = \frac{\partial C \dot{V}_{CK}}{\partial \underset{\sim}{V}_C}\,\delta \underset{\sim}{V}_C$$

TRANSFORMATION TO THE STATE-VARIATIONAL NETWORK.

Fig. 4

$$P(\underset{\sim}{p}) = \int_0^{t_f} \left\{ h(\underset{\sim}{x}[\underset{\sim}{p}, t], \underset{\sim}{p}, t) + \underset{\sim}{\lambda}^T(\dot{x} - f) \right\} dt$$

$$\dot{\underset{\sim}{\lambda}}^T = \underset{\sim}{\lambda}^T \frac{\partial \underset{\sim}{f}}{\partial \underset{\sim}{x}} + \frac{\partial h}{\partial \underset{\sim}{x}}$$

$$\frac{\partial P}{\partial \underset{\sim}{p}} = -\underset{\sim}{\lambda}^T(0) \frac{\partial \underset{\sim}{x}(0)}{\partial \underset{\sim}{p}} + \int_0^{t_f} \left[\frac{\partial h}{\partial \underset{\sim}{p}} - \underset{\sim}{\lambda}^T \frac{\partial \underset{\sim}{f}}{\partial \underset{\sim}{p}} \right] dt$$

Fig. 5

$$\hat{\underset{\sim}{I}}_{LK} = \text{COL}(\underset{\sim}{\lambda}^T{}_1, \underset{\sim}{\lambda}^T{}_2, \underset{\sim}{\lambda}^T{}_3, \lambda^T{}_4)$$

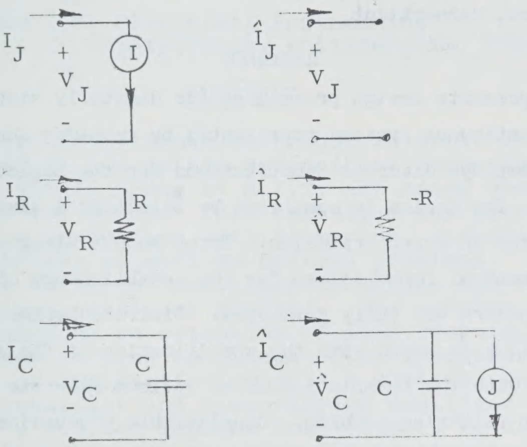
$$\hat{\underset{\sim}{V}}_{LK} = \text{COL}(\underset{\sim}{\lambda}^T{}_{13}, \underset{\sim}{\lambda}^T{}_{11}, \hat{V}_L, \underset{\sim}{\lambda}^T{}_{10})$$

$$\hat{\underset{\sim}{V}}_{TB} = \text{COL}(\underset{\sim}{\lambda}^T{}_5, \underset{\sim}{\lambda}^T{}_6, \underset{\sim}{\lambda}^T{}_7, \underset{\sim}{\lambda}^T{}_8)$$

$$\hat{\underset{\sim}{I}}_{TB} = \text{COL}(\underset{\sim}{\lambda}^T{}_{11}, \underset{\sim}{\lambda}^T{}_{12}, \underset{\sim}{I}^T{}_C, \underset{\sim}{\lambda}^T{}_9)$$

Correspondence between adjoint network and Lagrange multipliers.

Fig. 6


I_J
+
V_J
-
I
$\hat{I}_J$
+
V_J
-
I_R
+
V_R
-
R
$\hat{I}_R$
+
$\hat{V}_R$
-
-R
I_C
+
V_C
-
C
$\hat{I}_C$
+
$\hat{V}_C$
-
C
J

$$J = \frac{\partial h}{\partial V_C} - \hat{V}_C \frac{\partial C\dot{V}_C}{\partial V_C} + \hat{V}_J \frac{\partial I}{\partial V_C} + \hat{I}_E \frac{\partial E}{\partial V_C}$$

Transformation to the adjoint-variational network.

Fig. 7

PREDICTING RESPONSE ERROR OF DIGITALLY SIMULATED TRANSFER FUNCTIONS

JOSEPH S. ROSKO
United Aircraft Research Laboratories
East Hartford, Connecticut

ABSTRACT

This paper presents design procedures for digitally simulating stationary linear continuous systems represented by transfer functions. The first technique employs discrete substitutions for the Laplace integrating operator $(1/s)^n$. The design is completed by selecting a simulation interval which satisfies an error criterion. Error Models along with their associated mathematical formulations for the establishment of digital simulation error a priori are fully developed. Discrete system design using the second technique is begun with the specification of the sampling interval and proceeds with the formulation of an optimum discrete system as an equivalent Wiener filtering problem. Considerable discussion is also devoted to simulation with stochastic as well as deterministic excitations.

EXTENSION OF THE VOLTERRA FUNCTIONAL TECHNIQUE TO THE ANALYSIS OF TIME-VARYING NONLINEAR SYSTEMS

JAMES C. C. SU*
Moore School of Electrical Engineering
University of Pennsylvania, Philadelphia, Pennsylvania

ABSTRACT

The Volterra functional technique is still relatively unknown, though it has gained recognition among the Statistical Communication Theory Group at M.I.T. and among some authors in other parts of the world. This technique has proved particularly useful in connection with filtering, detection, and prediction problems as well as control problems. Heretofore, the use of the Volterra functional technique has been limited to time-invariant nonlinear systems. The purpose of this paper is intended as an extension of this technique to the analysis of time-varying nonlinear (TVN) systems subject to deterministic or stochastic inputs. A unified representation of both time-invariant and time-varying nonlinear systems is used and thus a Fourier-Laplace transform is secured. Many familiar relationships in the time-invariant case are carried over to the time-varying case. Cascaded systems analysis is considered in both domains. As much the same way as in the time-invariant case TVN systems can be characterized by TVN transfer functions. Statistical analysis is discussed; the autocorrelation function of the output is expressed in terms of the autocorrelation function of the input. Examples are given to illustrate the method.

For convenience of analysis, TVN systems are divided into three classes as follows:

$$\text{Class A:}\quad Z(D)x(t) + F(t;x, \dot{x},\cdots,x^{(n-1)}) = r(t) \tag{1}$$

$$\text{Class B:}\quad L(t;D)x(t) + F(x, \dot{x},\cdots,x^{(n-1)}) = r(t) \tag{2}$$

$$\text{Class C:}\quad L(t;D)x(t) + F(t;x, x,\cdots x^{(n-1)}) = r(t) \tag{3}$$

where Z(D) is a n-th order linear differential operator with constant coefficients, L(t;D) is a n-th order linear differential operator with time-varying coefficients, $D=d/dt$, $F(\cdot)$ is a nonlinear function, time-invariant or time-varying, and x(t) denotes the response of a system to an input r(t).

According to the Volterra functional technique, the total response x(t) to an input r(t) is assumed to be an infinite sum of $x_i(t)$,

$$x(t) = \sum_{i=1}^{\infty} x_i(t) \tag{4}$$

where

$$x_i(t) = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} k_i(t;\tau_1,\tau_2,\cdots,\tau_i) \prod_{j=1}^{i} r(t-\tau_j)d\tau_j \tag{5}$$

Equation (5) is obtained as follows. Substituting Eq. (4) into system equation [(1) or (2), (3)] and equating terms of equal order in r(t) gives a set of nonhomogeneous linear differential equations (d.e.) whose differential operators are exactly the same as the linear part differential operator of the original equation. In other words, solving a TVN d.e. is equivalent to solving a set of linear d.e. Then, writing these sets of nonhomogeneous d.e. in the convolutional integral form and substituting all previous terms into the terms which are of interest, the TVN Volterra functional represented by Eq. (5) is obtained.

---

*The author is now with Communication Lab., Communications Satellite Corp., Washington, D.C.

For Class B TVN systems in particular, the linear part system response to an input r(t) is given in an explicit form as

$$\chi_1(t) = \int_{-\infty}^{\infty} k_1(t;\tau)r(t-\tau)d\tau \tag{6}$$

and its Laplace transform is correspondingly

$$X_1 = K_1(t;s)R(s) = K_1^*(\lambda;s)R(s) \tag{7}$$

where Zadeh's linear system function and bi-frequency system function are respectively defined by

$$K_1(t;s) = \int_{-\infty}^{\infty} k_1(t;\tau)e^{-s\tau}d\tau \tag{8}$$

and

$$K_1^*(\lambda;s) = \int_{-\infty}^{\infty} K_1(t;s)e^{-\lambda t}dt = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} k_1(t;\tau)e^{-\lambda t-s\tau}dtd\tau \tag{9}$$

The TVN transform of Eq. (5) will be in the form

$$X_i = K_i^*(\underline{\lambda};\underline{s}) \prod_{j=1}^{i} R(s_j) \tag{10}$$

where $\underline{\lambda}$ and $\underline{s}$ denote the sets of multi-dimensional bi-frequency variables, and K_i^* is referred to "TVN transfer function."

In general, the time response can be obtained either by Eq. (5) or by taking the inverse one-dimensional Laplace transform of Eq. (10) after the association of all frequency variables, $\underline{\lambda}$ and $\underline{s}$, has been completed.

If the system is a time-invariant system, the terms involving time-varying parameters in the Volterra kernel k_i vanish and Eq. (5) reduces to

$$\chi_i(t) = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} h_i(\tau_1,\tau_2,\cdots\tau_i) \prod_{j=1}^{i} r(t-\tau_j)d\tau_j \tag{11}$$

which is a multi-dimensional time-invariant convolution integral

ACKNOWLEDGEMENT

The author wishes to express his sincere appreciation to Drs. Fred Haber and C. N. Weygandt for their invaluable advice, suggestions and discussions during the course of the study. The work was partly supported by the Moore School of Electrical Engineering, University of Pennsylvania, Philadelphia, Pennsylvania, and the National Science Foundation, Washington, D.C.

"VOLTERRA SYSTEM FUNCTION"
AND A CLASS OF NONLINEAR TIME-VARIANT SYSTEMS

S. NARAYANAN* and A. LAVI
Bell Telephone Laboratories, No. Andover, Massachusetts
Carnegie Tech., Pittsburgh, Pennsylvania

SUMMARY

This paper extends Zadeh's concept of system function developed for linear time-varying systems to a class of time-variant nonlinear systems. The Volterra system function is a multidimensional transform of the Volterra time-variant kernel. It is shown that this system function is physically meaningful and is especially useful in analyzing slowly varying frequency dependent systems with small nonlinearities. Each order system function satisfies a linear nonhomogeneous time-varying differential equation involving lower order system functions, and these can be solved approximately. Lastly, a class of systems called 'compatible system' is defined in terms of Volterra system function, and it is shown that the system function is useful in analyzing cascaded and feedback nonlinear compatible systems.

ABSTRACT

There is an increasing interest in nonlinear system analysis, especially with the advent and use of solid-state devices. Recently, for example, a frequency dependent nonlinear model for a transistor was analyzed, using time-invariant Volterra series representation. This paper considers a time-variant Volterra series, especially a transform method approach. The "Volterra system function" is defined as a transform of time-variant Volterra kernel. The physical significance of the system function, its use and a procedure for its determination are investigated.

In this paper, the N^{th} degree 'Volterra system function' of a system characterized by the N^{th} degree Volterra kernel is defined by

$$H_N(t;\lambda_1,--,\lambda_N) \prod_{i=1}^{N} k(\lambda_i,t) = \int_{i=1-N} d\tau_i h_N(t;\tau_1--\tau_N) \prod_{i=1}^{N} k(\lambda_i,\tau_i)$$

where $k(\lambda_i,\tau_i)$ is the i^{th} inverse transform kernel. It can be shown that the output of the system is simply the inverse of the product of the system function and the input transform, i.e., the inverse of

---

* Most of this work was done while the author was at Carnegie Tech., (1964-1965) and forms a part of his thesis, submitted in May, 1965.

$$H_N(t;\lambda_1, --,\lambda_N) \overset{N}{\underset{i=1}{\pi}} X(\lambda_i).$$

The physical significance of the Volterra system function is considered next. If two sinusoidal signals at frequencies f_1 and f_2 are impressed on a system characterized by the Volterra kernel $h_2(t;\tau_1,\tau_2)$, the steady state gain and phase at the product frequency f_1-f_2 is given by the Volterra system function $H_2(t;j\omega_1, -j\omega_2)$. This could, for example, represent the time-varying (due to time-varying temperature) intermodulation distortion of a transistor at the second order product frequency.

To study the frequency variations introduced by the time-varying parameters, a new system function, called 'N+1 Volterra system function', can be defined. This is the Fourier transform of the N^{th} order Volterra system function $H_N9t;\underline{j\omega_i})$ an abbreviation for $H_N9t;j\omega_1, --,j\omega_N)$ and is denoted by $H_N(j\omega;j\omega_i)$. As a practical example, consider a cascade of solid state amplifier and a coaxial cable which is subjected to a time-varying signal of bandwidth W_s. It can be shown, for example, that the second harmonic product signal at f_1-f_2 will be spread from $f_1-f_2-W_s$ to $f_1-f_2+W_s$, and the value of the output transform at $\omega_1-\omega_2+\omega$ is given by $H_2(j\omega;j\omega_1,-j\omega_2)$. In general, $H_N(j\omega;\underline{j\omega_i})$ depends on 'N+2' frequencies.

The Volterra system function $H_N(t;\underline{\lambda})$ can be determined directly from the differential equation characterizing the system. It can be shown that each order system function satisfies a linear time-varying nonhomogeneous differential equation which involves lower order system functions. Thus, each order system function can be successively solved approximately. This technique is especially applicable if the system is slowly varying and contains small nonlinearities.

A time-varying nonlinear system is defined as 'compatible' if $H_N(t;\underline{\lambda})$ is independent of t for a particular transform kernel $k(\lambda_i,t)$. It is shown that a linear time-varying system to be compatible in cascade with a nonlinear system must, in general, satisfy an additional requirement. Two specific systems, one compatible for a transform kernel of the form $e^{\lambda i \int_0^t g(v)dv}$, and the other compatible for mellin transform kernel $t^{-\lambda i}$ are considered. It is shown that the Volterra system function is useful in analyzing cascaded and feedback nonlinear 'compatible' systems.

MULTI-POINT DESIGN OF TIME-VARYING FEEDBACK CONTROLLERS

D. M. Salmon
Stanford Research Institute
Menlo Park, California

P. Kokotović
University of Illinois, Urbana, Ill.
and M. Pupin Research Institute,
Belgrade, Yugoslavia

ABSTRACT

The paper develops a procedure for the design of time-varying controllers for non-linear or linear time-varying plants, $\dot{x} = f(x,u,v,t)$, $x^o = \varphi(t_o,v)$, where v is a time-invariant parameter vector. The design is based on the fact that a state feedback controller defined by an m x n matrix C(t) is capable of generating the optimal controls at n+1 points of the parameter domain. The design requires the computation of the optimal control at "representative points" and the inversion of an n x n time-varying matrix. When the amount of computation in "multi-point design" is prohibitive, the paper suggests the "multi-point neighborhood design" whereby the obtained control is optimal at less than n+1 points, but is also optimal in a wider neighborhood of these points.

INTRODUCTION

It is seldom possible to specify accurately all characteristics of a plant to be controlled. Instead of fixing all plant parameters at hypothetical nominal values it is often more realistic to admit that only a domain of their possible values is known. When the parameters can be measured and when the optimal control is known for each set of parameter values, then, at least in theory, a controller can be constructed which measures the parameters and applies the optimal control accordingly. Such a controller will be referred to as "the ideal controller." Although the ideal controller yields the best possible system performance, it is inherently complex and expensive to realize. Therefore, there is strong motivation for attempting to find controllers which are easy to realize physically yet approximate the performance of the ideal controller.

Most methods for approximating the ideal controller may be classified as local or global. Normally local methods are based upon a Taylor series expansion of an optimal control at nominal values of the plant parameters. In contrast, global methods involve the system performance over the entire parameter domain. A typical deterministic approach is to mini-maximize either the performance functional or a performance-sensitivity derived from it [1]. This paper proposes a design method which may be considered as intermediate between local methods and mini-max procedures.

For a comparison of different approaches to controller design at least three factors are important. These are the achieved system performance, the amount of computation needed for the design, and the complexity of the resulting controller. Local methods using the first order approximation need a moderate amount of computation and the resulting controller is practically realizable. Unfortunately, for large parameter variations the system performance often becomes unacceptable. It is doubtful that the improvement gained by higher order approximations is sufficient to justify the rapid rise of controller complexity. Minimax methods can be employed to design a practically realizable controller which approximates the performance of an ideal controller over the entire parameter domain. However, especially in the case of time-varying controllers, computation is difficult.

The "multi-point design" method proposed in this paper attempts to achieve a performance superior to that achieved by first order local methods, with less computation than is required for mini-max methods [1]. A multi-point optimal controller guarantees optimality at "representative" points of the parameter domain. Generally the number of points is equal to the number of time-varying parameters assumed in the controller structure. If a linear state feedback controller is assumed the number of points can be as high as n+1 for an n-th order plant. Two versions of the method are presented in the paper and illustrated by examples.

MULTI-POINT OPTIMALITY

Let the right-hand side of the plant equation as well as the initial state depend on a time-invariant or slowly varying r-vector v,

$$\dot{x} = f(x,u,t,v), \quad x \quad \text{n-vector}; \quad u \quad \text{m-vector}; \tag{1}$$

$$x(t_o) = \varphi(v). \tag{2}$$

The unknown parameter vector is neither adjustable nor measurable but is known to take on values from an admissible domain V. It is assumed that for any one value of $v \in V$, there is a unique bounded optimal control u which minimizes the performance functional

$$J(u) = \int_{t_o}^{t_f} L(x,u,t,v)dt, \tag{3}$$

subject to the side condition (1), with the initial state (2) and the final state $x(t_f)$ free. Let $u(t,v)$ be the parametric family of optimal controls generated by the ideal controller. The corresponding family of trajectories is denoted by $x(t,v)$.

Consider a collection Z^{n+1} of n+1 points $v^o, v^1, \ldots, v^n$ of the domain V. A parametric family $\tilde{u}(t,v)$ is called a "multi-point optimal control" on the collection Z^{n+1} if the equalities

$$\tilde{u}(t,v^i) = u(t,v^1), \quad i = 0,1,2,\ldots,n; \tag{4}$$

hold for all $t \in [t_o,t_f]$ except possibly for a countable number of values of t. The family of trajectories corresponding to $\tilde{u}(t,v)$ is denoted by $\tilde{x}(t,v)$. It is assumed that the functions f, φ, and L are sufficiently smooth and that the solution of (1) is unique for all $v \in V$ and all u, and hence

$$\tilde{x}(t,v^i) = x(t,v^i), \quad i = 0,1,2,\ldots,n. \tag{5}$$

The following development of multi-point design assumes a linear state feedback controller. The development for other controller structures is similar. Let the controller equation be

$$\tilde{u}(t,v) = c^o(t) + C(t)\tilde{x}(t,v), \tag{6a}$$

where C(t), an m x n matrix, and $c^o(t)$, an m-vector, are to be determined. An equivalent form of (6a) is

$$\tilde{u}(t,v) - u(t,v^o) = C(t)[\tilde{x}(t,v) - x(t,v^o)], \tag{6b}$$

where

$$c^o(t) = u(t,v^o) - C(t)x(t,v^o). \tag{6c}$$

The substitution of v^i for v in (6b) generates n linear m-dimensional vector equations

$$u(t,v^i) - u(t,v^o) = C(t)[x(t,v^i) - x(t,v^o)]; \qquad i = 1,2,\ldots,n \tag{7}$$

for the mn elements of C(t). A compact form of the solution C(t) is obtained if the "multi-point control matrix" U(t) and the "multi-point trajectory matrix" X(t) are introduced,

$$U(t) = [u^1 \vdots u^2 \vdots \ldots \vdots u^n]; \quad X(t) = [x^1 \vdots x^2 \vdots \ldots \vdots x^n], \tag{8}$$

where

$$u^i = u(t,v^i) - u(t,v^o); \quad x^i = x(t,v^i) - x(t,v^o). \tag{9}$$

Then, by substituting (8) and (9) in (7) and postmultiplying by $X^{-1}(t)$,

$$C(t) = U(t)X^{-1}(t). \tag{10}$$

The computational procedure has two parts. First the representative points are selected and the functions (9) are calculated. Then C(t) is computed using (10).

There are several situations which may cause the matrix X to be singular, not all of which prevent the application of the method. If the rows of X are linearly dependent on a finite interval of time it follows that the system order has been overstated at n. If the columns of X, that is the trajectories are linearly dependent on a finite interval of time, a change can be made in the representative points. It can happen that, from $t = t_c$ on some of the trajectories $x(t,v^i)$ almost coincide, and that the inversion of X(t) is practically impossible. This situation is more profitable than not. The closeness of trajectories for different v^i's indicates that the system is insensitive to v, and that it is irrelevant which of the corresponding controls $u(t,v^i)$ will be applied. As the Example 1 shows, such a situation typically occurs when the system is forced to follow a desired trajectory. It can also occur that X is singular at a finite number of time instants and that some of the elements of C(t) become infinite. Finite values for the elements of C(t) must be adopted at those instants. This should not lead to significant deterioration in performance since the consequent error in the control is finite and lasts only for a small time interval.

Some properties of multi-point optimal systems may be illustrated by defining the performance-sensitivity

$$S(v) = J(\tilde{u}(t,v)) - J(u(t,v)). \tag{11}$$

The non-negative function S(v) is a measure of the deviation in actual system performance from the performance with the ideal controller. Evidently S(v) is zero at the representative points v^i. Moreover, the assumed smoothness of the functions involved in (1), (2) and (3) provides that S(v) is smooth too, and hence attains a minimum at every representative point. To summarize, for $v = v^i$, $i = 0,1,\ldots,n$, the following relations hold

$$S(v) = 0; \quad \frac{\partial S}{\partial v} = 0; \quad \frac{\partial^2 S}{\partial v^2} \text{ positive definite.} \qquad (12)$$

The curves S' and S" in Fig. 1 indicate two typical forms of S(v) which may occur in a second-order system with a scalar parameter v. Since n = 2 there are three representative points v^o, v^1, v^2 where both S' and S" must be zero and have minima. It can be expected that for a class of "well-behaved" plants the multi-point optimality implies a performance sensitivity similar to S'. The multi-point design method is particularly suitable for such plants.

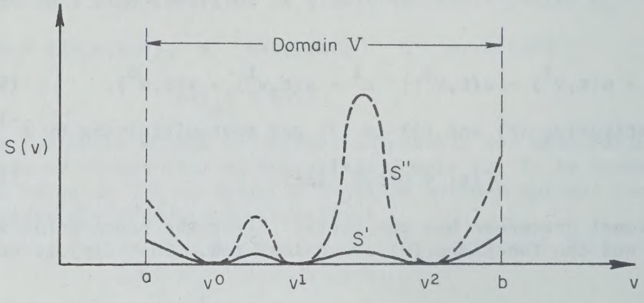


Fig. 1

If the sensitivity S(v) is similar to S" this implies that the collection Z^{n+1} is not "representative" of the domain V. It can be considered as the first guess Z_I^{n+1} only. The result of the first guess indicates the direction for the second choice Z_{II}^{n+1}. For example, the curve S" in Fig. 1 suggests that in Z_{II}^{n+1} points v^1 and v^2 should be closer to each other and both closer to b. Although more sophisticated algorithms for iterative choices of Z_I^{n+1}, Z_{II}^{n+1}, etc, can be invented, the main assumption of the method is that not more than two or three steps are necessary to get a satisfactory form of S(v). If this is not the case, then the number of representative points must be made greater than n+1. The inclusion of non-linear or dynamic terms in the controller equation (6) would permit the satisfaction of more than n+1 conditions of the form (4).

EXAMPLE 1. Given the plant

$$\dot{x}_1 = x_2; \quad \dot{x}_2 = -.25x_1 - .2x_2 + 10u, \qquad (13)$$

and the performance index

$$J = \tfrac{1}{2}\int_0^{10} [.068(y_1 - y_1^d)^2 + .021(y_2 - y_2^d)^2 + .021(5y_2)^{16} + u^2]dt \qquad (14)$$

where $y_1 = 2x_1$; $y_2 = x_2$ and y_1^d and y_2^d define the desired trajectory

$$y_1^d = \tfrac{1}{2}(1 + \cos\frac{\pi t}{10}), \quad y_2^d = -\frac{\pi}{40}\sin\frac{\pi t}{10}. \qquad (15)$$

The term $(5y_2)^{16}$ which makes the performance index non-quadratic and greatly complicates the design is introduced to penalize the controller if the constraint $|y_2(t)| \leq 2$ is violated. The unknown parameter vector v in this problem is the initial state, $x(0) = v$, and the admissible domain V is defined by

$$.5 \leq v_1 \leq .75; \quad 0 \leq v_2 \leq .2 \; . \tag{16}$$

The above problem, which occurred in speed control of a rotary shear, has been treated in [2]. There, the point B in Fig. 2 ($v_1 = .75$; $v_2 = .2$), is taken as "nominal" and a "nominal-optimal" controller is designed. Here with the same controller structure, better performance is obtained.

The controller equation (6a) becomes

$$\tilde{v}(t,v) = c^o(t) + c_1(t)\tilde{x}_1(t,v) + c_2(t)\tilde{x}_2(t,v). \tag{17}$$

A version of Newton-Kantorovich method [3] was used for the calculations of $u(t,v^i)$, $x_1(t,v^i)$, $x_2(t,v^i)$; $i = 0,1,2$. Then the unknowns $c^o(t)$, $c_1(t)$, $c_2(t)$ were computed from the system of three linear equations

$$\begin{aligned} c^o(t) + x_1(t,v^o)c_1(t) + x_2(t,v^o)c_2(t) &= u(t,v^o) \\ c^o(t) + x_1(t,v^1)c_1(t) + x_2(t,v^1)c_2(t) &= u(t,v^1) \\ c^o(t) + x_1(t,v^2)c_1(t) + x_2(t,v^2)c_2(t) &= u(t,v^2). \end{aligned} \tag{18}$$

The choice of representative points is not critical in this example, since any three widely separated points yield a good controller. The result obtained with the collection Z^3 of points $R_1^3 = (.5;.1)$, $R_2^3 = (.7;0)$, $R_3^3 = (.7;.2)$, is shown in Fig. 2, where the curve d is the desired trajectory. The trajectories plotted correspond to the most unfavorable points A and B of the domain V. It should be noted that both trajectories A and B are virtually indistinguishable from the ideal trajectories. For all other points in V, the performance is as good or better. In contrast the nominal-optimal controller gives acceptable performance only in the neighborhood of B. The curve N in Fig. 2 shows the trajectory at the point A with the nominal-optimal controller.

It is apparent from Fig. 2 that the optimal trajectories are almost identical from $t = 3.5$ to $t = 10$ regardless of the value of v. Consequently, the determinant of the system (18) tends toward zero as the time progresses and (18) can be solved only on the interval $0 \leq t \leq 3.5$. However, it is not necessary to solve (18) for $3.5 < t \leq 10$ since the same control is required at all of the representative points.

MULTI-POINT NEIGHBORHOOD OPTIMALITY

The multi-point design as described requires knowledge of the optimal controls and optimal trajectories at n + 1 representative points. If the computation becomes involved the designer may be forced to base the design upon less than n + 1 representative points. If the number of points is $h < n + 1$, then $n + 1 - h$ additional conditions are needed to determine the feedback matrix C(t). If the calculation of the h optimal controls is done by a version of the Newton-Kantorovich method, the additional conditions are provided as a by product. The procedure will be called the multi-point neighborhood design.

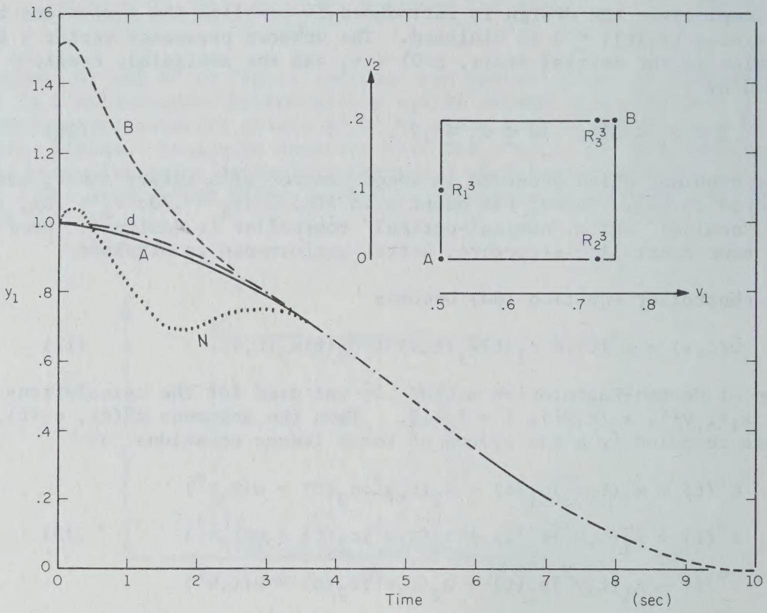


Fig. 2

For the sake of clarity the idea will be explained for the case of third order plant with two representative points, v^o and v^1. Then $3 + 1 - 2 = 2$ additional conditions are needed. Without loss of generality it is assumed that the dimensionality of v is $r = 2$. Suppose now that a special choice of two additional representative points v^2 and v^3 is made as follows

$$v^2 = v^o + \begin{bmatrix} w_1 \\ 0 \end{bmatrix} ; \quad v^3 = v^o + \begin{bmatrix} 0 \\ w_2 \end{bmatrix} . \qquad (19)$$

The scalars w_1 and w_2 are small and define a two-dimensional neighborhood of the point v^o. If they tend to zero, the substitution of (19) into (9) gives

$$\lim_{w_1 \to o} \frac{u^2}{w_1} = \frac{\partial u(t,v)}{\partial v_1} ; \quad \lim_{w_1 \to o} \frac{x^2}{w_1} = \frac{\partial x(t,v)}{\partial v_1} ;$$

$$\lim_{w_1 \to o} \frac{u^3}{w_2} = \frac{\partial u(t,v)}{\partial v_2} ; \quad \lim_{w_2 \to o} \frac{x^3}{w_2} = \frac{\partial x(t,v)}{\partial v_2} ; \qquad (20)$$

where the partial derivatives are evaluated at the point $v = v^o$. The partial derivatives (20) are called optimal sensitivity functions and are denoted by

$$\theta_k^o = \theta_k^o(t) = \left.\frac{\partial u(t,v)}{\partial v_k}\right|_{v=v^o} \quad ; \quad \gamma_k^o = \gamma_k^o(t) = \left.\frac{\partial x(t,v)}{\partial v_k}\right|_{v=v^o} \qquad (21)$$

where the superscript "o" indicates that they are calculated at v^o. It is clear from (7), (8) and (9) that multiplication of the i-th columns of matrices U and X by the same scalar does not influence the matrix C(t). Therefore instead of U and X the matrices

$$U_w(t) = [u^1 \vdots \frac{u^2}{w_1} \vdots \frac{u^3}{w_2}] \; ; \quad X_w(t) = [x^1 \vdots \frac{x^2}{w_1} \vdots \frac{x^3}{w_2}] \; ; \qquad (21)$$

can be used in (10). Then, as w_1 and w_2 tend to zero,

$$U_w(t) \rightarrow U_s(t) = [u^1 \vdots \theta_1^o \vdots \theta_2^o]; \quad X_w(t) \rightarrow X_s(t) = [x^1 \vdots \gamma_1^o \vdots \gamma_2^o], \qquad (22)$$

and the matrices U_s and X_s can be substituted in (10), to give the expression for the multi-point neighborhood controller $C_s(t)$,

$$C_s(t) = U_s(t) X_s^{-1}(t) \; . \qquad (23)$$

In the above example a two-dimensional neighborhood of the point v^o provided two additional conditions. In the same example it would have been possible to select other forms of the multipoint neighborhood. Two one-dimensional neighborhoods of points v^o and v^1 or alternatively, a two-dimensional neighborhood of the point v^1 could have been chosen. In any case the corresponding optimal sensitivity functions would appear in matrices U_s and X_s .

The performance of a multi-point neighborhood optimal controller can be estimated from the fact that at the point whose complete neighborhood is taken S(v) and its first, second, and third derivatives are zero. In other words, at the price of having fewer points to represent the domain V, better approximation at some of the points is achieved. The first step in the design of a multi-point neighborhood controller is computationally simpler since it is much easier to find the optimal sensitivity functions $\theta_k(t)$, $\gamma_k(t)$ at a point where the optimal control is known, than to calculate optimal controls for new points. The optimal sensitivity functions are computed by the straightforward solution of the optimal sensitivity equations which are derived in the paper [4] in these Proceedings. The trajectory sensitivity $\gamma_k(t)$ is computed from the formula

$$\gamma_k(t) = \zeta_k(t) \frac{\partial \varphi}{\partial v} + \sigma_k(t) \; . \qquad (24)$$

The vectors $\zeta_k(t)$ and $\sigma_k(t)$ represent the k-th columns of matrices $\zeta(t)$ and $\sigma(t)$ and for their computation sensitivity equations (45) and (46) of [4] have to be solved. Then the control sensitivity $\theta_k(t)$ can be obtained from formula (35) of [4].

EXAMPLE 2. For the plant (13), performance index (14) and controller structure (17), two choices of representative points and neighborhoods were made. Fig. 3 shows trajectories obtained with two

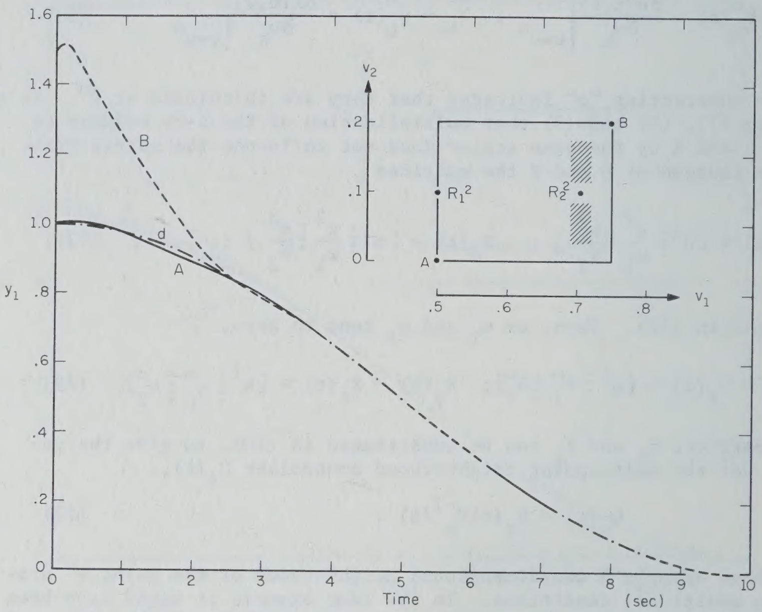


Fig. 3

representative points, R_1^2 and R_2^2. The third condition for determining C(t) and $c^o(t)$ was obtained from the sensitivities, $\partial u/\partial v_2$, $\partial x/\partial v_2$, at R_2^2. In other words a partial neighborhood has been taken of R_2^2. Fig. 4 shows trajectories obtained with one representative point R_1^1. The two additional conditions were obtained from the complete neighborhood defined by $\partial u/\partial v_1$, $\partial u/\partial v_2$, $\partial x/\partial v_1$, $\partial x/\partial v_2$, evaluated at R_1^1. As seen from Fig. 4, the least satisfactory performance is obtained in the latter case which also required the least computation. For this example the best compromise between computation and performance appears to be given by two representative points with one partial neighborhood (see Fig. 3).

CONCLUSION

Taken together, the methods of multi-point and multi-point neighborhood design allow flexibility in compromising between system performance and computation required to determine the controller. The methods are based on the assumption that the behavior of a system with unknown parameters can be sufficiently well represented by its behavior at a small number of points in the parameter domain.

One possible extension would be to introduce additional representative points and to satisfy the resulting over-determined equations for C(t) in the sense of a pseudo-inverse [5]. Other possibilities are to increase the number of representative points by adding non-linear feedback terms or by introducing dynamics into the controller.

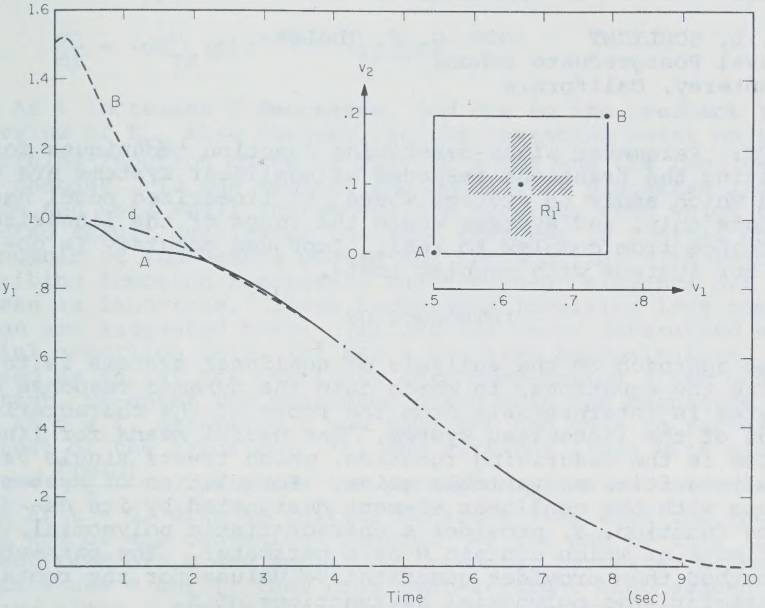


Fig. 4

REFERENCES

[1] D. M. Salmon, "Minimax Controller Design," Coordinated Science Laboratory Report R-358, University of Illinois, July 1967.

[2] P. M. DeRusso, R. J. Roy and M. C. Close, "State Variables for Engineers," John Wiley and Sons, Inc., 1965.

[3] J. V. Breakwell, J. L. Speyer and A. E. Bryson, "Optimization and Control of Nonlinear Systems using the Second Variation," J. SIAM Control, Ser. A, Vol. 1, No. 2, 1963.

[4] P. Kokotović, J. Heller, P. Sannuti, "Obtaining and Applications of Optimal Sensitivity Functions," Proceedings Fifth Allerton Conference, October, 1967, University of Illinois, Urbana, Illinois.

[5] T. N. E. Grenville, "The Pseudo Inverse of a Rectangular Matrix and its Applications to the Solution of System of Linear Equations," SIAM Review, Vol. No. 1, January 1959.

ACKNOWLEDGMENT

The work reported in this paper has been supported by JSEP under Army Contract DAAB-07-67-C-0199 and Air Force Contract AFOSR 931-67.

TRANSIENT RESPONSE OF NONLINEAR SYSTEMS

D. L. SCHLICHT G. J. THALER
Naval Postgraduate School
Monterey, California

ABSTRACT: Parameter plane-describing function techniques for calculating the transient response of nonlinear systems are developed which apply to systems where the linearized model has real roots only, and systems where the roots of the linearized model change from complex to real. Improved accuracy is obtained for systems with complex roots.

INTRODUCTION

One approach to the analysis of nonlinear systems is to linearize the equations, in which case the dynamic response of the system is interpretable from the roots of the characteristic equation of the linearized system. One useful means for linearization is the describing function, which treats single valued nonlinearities as variable gains. Formulation of system equations with the nonlinear element designated by its describing function, N, provides a characteristic polynomial, the coefficients of which contain N as a parameter. The parameter plane method then provides quantitative values for the roots of the characteristic polynomial as functions of N.

Previous investigations[1,2,3] have used the parameter plane with describing functions to predict limit cycle performance of nonlinear systems, and to predict the transient response of nonlinear systems for which the linearized model contained a dominant pair of lightly damped complex roots. This paper is an extension of these studies. It is concerned with the transient response of systems with dominant real roots, with dominant complex roots without restriction to the amount of damping, and with higher order systems in which the dominance shifts from complex roots to real roots during the transient as a consequence of the nonlinear characteristics. Techniques are developed which provide rapid and reasonably accurate construction of the transient response (from initial conditions), and which permit interpretation of design requirements.

FIRST ORDER SYSTEMS

Consider the system of Fig. 1a, in which the nature of the nonlinearity, N, is not specified. The characteristic equation of this system is

$$s + \alpha + KN = 0 \qquad (1)$$

and the parameter plane curves for the roots is given on Fig. 1b. Note that this family of curves is independent of the nature of the nonlinearity. The describing function locus shown on Fig. 1b applies for a saturation type nonlinearity with saturation level $E_s = \mp 4.0$. As seen from the curve, if the initial condition is $E_{IN}(0) = 40$, $\sigma = -0.626$, then the initial solution for the linearized system is

$$C = E_{IN}(0)\,\epsilon^{-\sigma t} = 40\epsilon^{-.626t} \tag{2a}$$

$$\frac{dc}{dt} = -\sigma E_{IN}(0)\,\epsilon^{-\sigma t} = -25.04\epsilon^{-.626t} \tag{2b}$$

As t increases C decreases, and due to the feedback loop the value of E_{IN} also changes, so the operating point on the describing function curve of Fig. 1b moves and the root value also changes. If one applies the concept of piecewise linearization and uses small increments the accuracy with which the computed result duplicates the true response is limited only by the number of increments chosen and the accuracy with which the describing function represents the nonlinear element; but the process is laborious. Three techniques involving less computation are suggested here: (a) use the roots determined at the initial condition only; (b) use a two step approximation, i.e., proceed as in (a) but terminate this computation after one time constant, re-enter the parameter plane with the magnitude evaluated at $t = \tau$, determine a new root and use this to represent the rest of the transient; (c) use straight lines with slope $dc/dt = \frac{E_{IN}(0)}{\tau}$; i.e., use the parameter plane to determine $\tau = \frac{1}{\sigma}$ at $t = 0$, draw an initial straight line with the indicated slope and of any convenient length. At the end of this line segment read off a new amplitude, re-enter parameter plane to evaluate new σ and τ, then draw next straight line segment at the new slope. Repeat as needed. Typical results are illustrated on Fig. 2, and are seen to be reasonably accurate. Similar results have been obtained[4] for systems in which the nonlinearity was dead zone and also for systems in which the nonlinearity was a combination of saturation and dead zone.

SECOND ORDER SYSTEMS

A stable second order linear system may have real roots or complex roots. A nonlinear second order system, after linearization, is representable by moving roots which may be always real, or always complex, finally roots which start as complex roots may become real roots during the transient (or vice versa). All three cases are considered here.

A. Roots Always Real

Consider the system of Fig. 3a, in which $\alpha = 2.4$ so that the roots of the linearized systems are always real as shown by the describing function curve on the parameter plane of Fig. 3b. The transient response of this system from initial conditions is

$$C = X\epsilon^{-\sigma_1 t} + Y\epsilon^{-\sigma_2 t} \tag{3}$$

where X is positive and Y is negative for σ_1 dominant. The approximating techniques used for the first order case are applicable here (except for the straight line technique) but leave something to be desired in accuracy in addition to requiring considerable computational labor. A modified technique using straight lines only has proven quite satisfactory.

The solution of eqn. 3 is the sum of two exponential terms, each of which can be represented by a straight line (on C vs t coordinates) with slope determined by the root value. When one root is appreciably smaller than the other and can be said to dominate, then for t greater than the time constant of the dominant root the contribution of the non-dominant root can be neglected. A graphical computation then consists of evaluating the initial roots from the describing function locus on the parameter plane, evaluating X (in eqn. 3) by residues and constructing a straight line segment from $C = X$ (at $t = 0$) with slope determined by the root value. This line is extended for some specified distance and terminated, the endpoint being a point on the transient response curve. From this point the signal amplitude is determined and the describing function locus entered to evaluate the new root locations. A new value of X (eqn. 3) is computed using residues, etc. This is repeated until the response is computed. Fig. 4 gives a typical example for the system described by Fig. 3, for which the nonlinearity is saturation. Additional results for dead zone, and for dead zone plus saturation were obtained[4] but are not included here.

An important point in the computation using this technique is the choice of a rule for terminating the straight line segments. Obviously somewhat different results will be obtained for different rules, and what is desired is a rule that provides reasonably accurate results with minimum labor. The following rule has been found to work well: with "step size" defined as the change in output (c) value after which the roots are reevaluated

a) for a saturation nonlinearity with saturation level E_s, choose a step size $E_s/2$.

b) for a deadzone nonlinearity with deadzone E_D (where E_D is one half of the total deadzone width) choose a stepsize $3E_D/2$.

c) for a nonlinear element with both saturation and deadzone, choose a step size $(\frac{E_s - E_D}{2})$.

It should also be noted that some alteration of the step size may become necessary during the computations, especially as the transient approaches steady state ($t \to \infty$).

B. Roots Always Complex

The philosophy and graphical techniques given in reference 3 apply here and will be used with minimum explanation. Improvements reported here are twofold. Since the solution of a second order linear differential equation with complex roots is

$$C(t) = A \left[\frac{1}{(1-\zeta^2)^{\frac{1}{2}}}\right] \epsilon^{-\zeta\omega_n t} \sin(\omega t + \psi) \qquad (4)$$

where $\omega = \omega_n(1-\zeta^2)^{\frac{1}{2}}$

$$\sigma = \zeta\omega_n$$

$$\psi = \sin^{-1}\frac{\omega}{\omega_n} = \cos^{-1}\frac{\sigma}{\omega_n}$$

The envelope of the transient response starts at $C(0)=A/(1-\zeta^2)^{\frac{1}{2}} = A\omega_n/\omega$ instead of $C(0) = A$. For example, the system of Fig.5a, with parameter plane and describing function curves as on Fig. 5b, has $C(0) = 23$ for $A = 20$, and $\zeta\omega_n = .5$ so that the envelope of the transient response on Fig. 6 starts at $C(0) = 23$, $t = 0$, with slope determined from $dc/dt = -C(0)/\tau = -C(0)\sigma$. The envelope is drawn with straight line segments, the first segment being constructed as indicated and extended to a suitable length. To determine the second segment the amplitude, C, is read from the graph at the termination of the first segment, then the parameter plane is entered with this value of C (on the describing function curve) and new root values are read off which define the slope of the second segment.

The transient response of the system is bounded by the envelope thus constructed, and the points at which the transient response is zero are computed by graphical integration of the phase relationship, i.e., since $C(t) = 0$ only for $\sin(\omega t + \psi)=0$

$$N\Pi = \psi + \int_0^t \omega\, dt \qquad (5)$$

and the values of t which make the right hand side of Eqn. 5 exactly $N\Pi$ are determined by plotting ω (determined from the parameter plane for a given amplitude on the describing function scale) vs t (determined from the transient envelope for the same amplitude) and graphically evaluating[3] the area under the curve. Additional points not available with the techniques of reference 3 may be estimated as follows:

a) Choose t and evaluate the envelope amplitude C' from the envelope plot.

b) For the same t evaluate $\int \omega\, dt$ from the plot of ω vs t.

c) Evaluate $\phi = \psi + \int_0^t \omega\, dt$

d) The point on the transient curve is given by $C(t) = C' \sin\phi$.

The results shown in Fig. 6 indicate the accuracy of the method. Similar results were obtained[4] for heavily damped systems.

C. Complex Roots Changing to Real Roots

The technique to be used here is a simple combination of those given in the two preceding sections. Consider the system of Fig. 7a and its parameter plane on Fig. 7b. For an initial condition of amplitude 20 the roots are complex and the envelope as well as the transient curve itself are constructed as previously indicated. When the roots become real due to decreasing amplitude, additional points on the transient curve are defined using the straight line technique. The results are shown on Fig. 8 and are seen to be reasonably accurate.

HIGHER ORDER SYSTEMS

When second order dominance can be assumed for a higher order system, or when the parameter plane is constructed in sufficient detail to show that such dominance exists, then the preceding methods apply directly. Fig. 9a shows a fourth order system and Fig. 9b shows the parameter plane and describing function for one pair of roots (assumed dominant). Assuming a step input and using the previously indicated constructions the response of Fig. 10 is obtained. Note that the steady state portion of the calculated curve is an "estimate", i.e., there is no known method for determining how far the last loop of the oscillating curve actually penetrates the deadzone before the stored energy is exhausted.

COMMENTS

This study has resulted in relatively simple means for calculating the transient response of a large number of nonlinear systems. Some means (preferably computer programs) are required for evaluating the parameter plane curves and the describing function of the nonlinearity, but once these have been used the transient response is obtained readily and accurately. The extension of these results to design has not been attempted in any formal manner, but inspection of the parameter plane plot permits an estimate of the direction in which the α parameter should be changed to increase (or decrease) the damping. In like manner it should be possible to estimate where the describing function locus should be placed on the parameter plane to obtain desired response characteristics, and this estimate should be interpretable in terms of the parameter values required to obtain the desired conditions.

REFERENCES

1. SILJAK, D.D. Analysis and Synthesis of Feedback Control Systems in the Parameter Plane III - Nonlinear Systems. IEEE Transactions on Applications and Industry. Nov. 1964.

2. WILSON, P.L. Use of Mitrovic's Method in Predicting Limit Cycles with One and Two Gain Variable Nonlinearities. MS Thesis, U.S. Naval Postgraduate School. 1964

3. PELLEGRINI, C.A; H.M. YOCKEY; G.J. THALER. System Performance of Control Systems with Two Nonlinearities. Proceedings, Fourth Annual Allerton Conference, Univ. of Illinois. 1966.

4. SCHLICHT, D.L. The Transient Response of Nonlinear Systems Using the Parameter Plane, Master's Thesis, Naval Post-Graduate School, 1967.

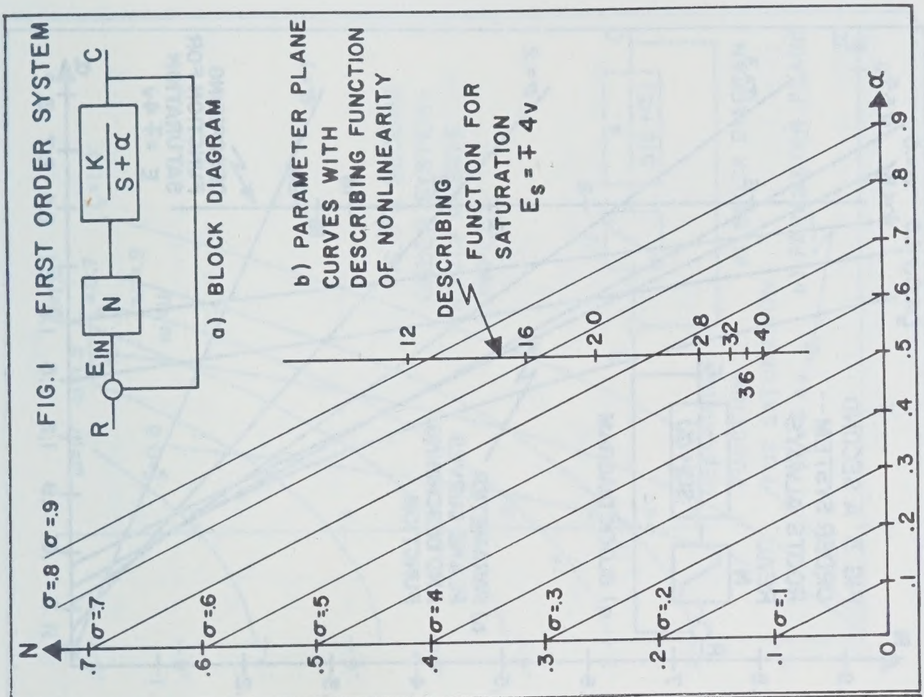


FIG. I FIRST ORDER SYSTEM

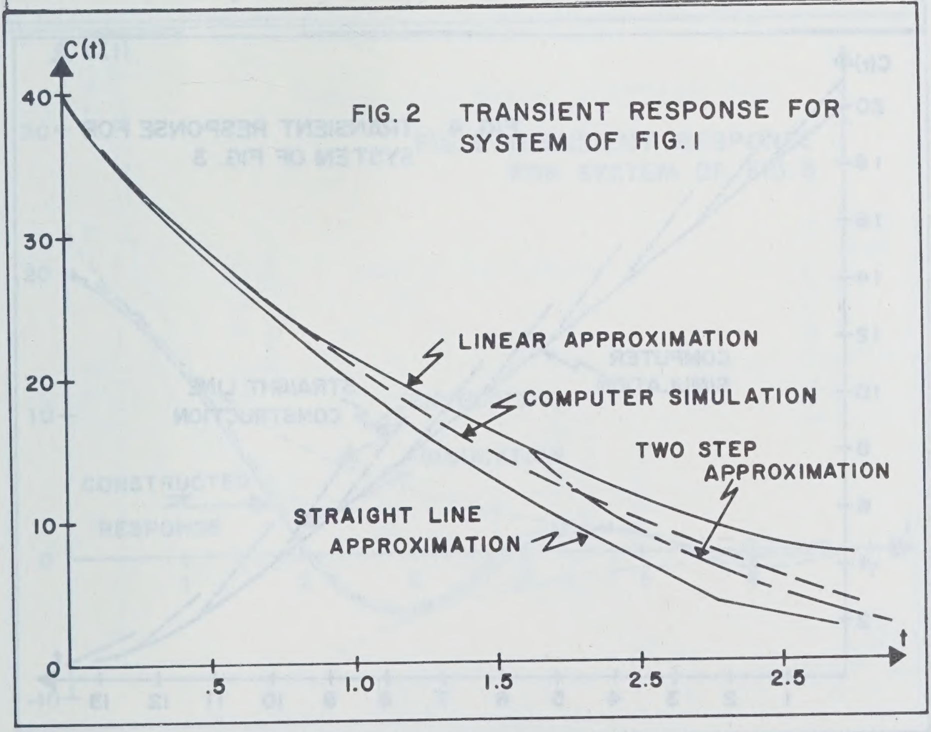


FIG. 2 TRANSIENT RESPONSE FOR SYSTEM OF FIG. I

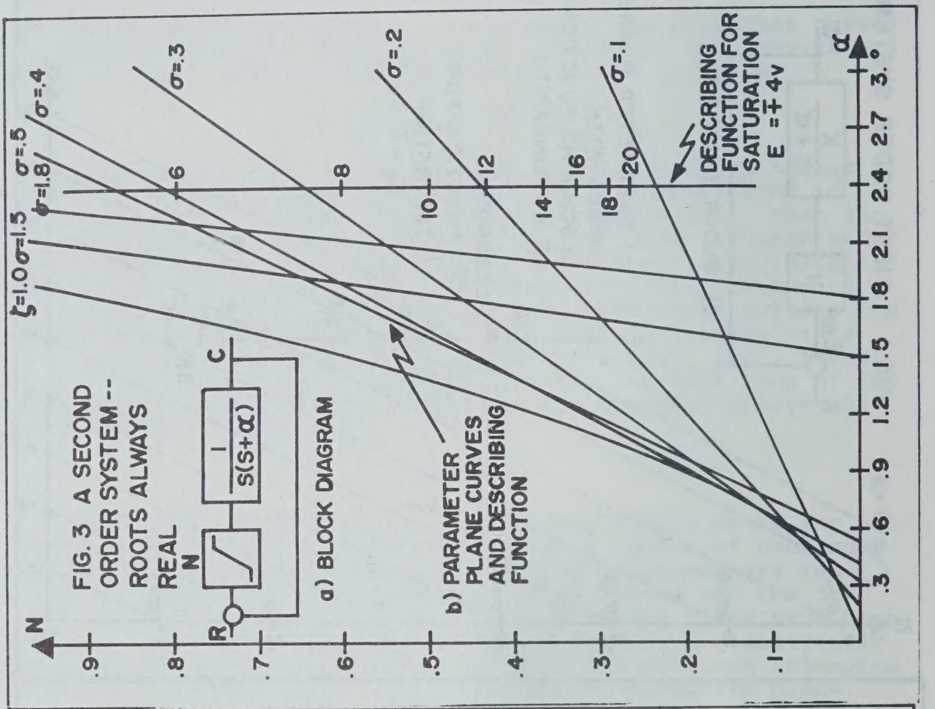


FIG. 3 A SECOND ORDER SYSTEM -- ROOTS ALWAYS REAL

a) BLOCK DIAGRAM

b) PARAMETER PLANE CURVES AND DESCRIBING FUNCTION

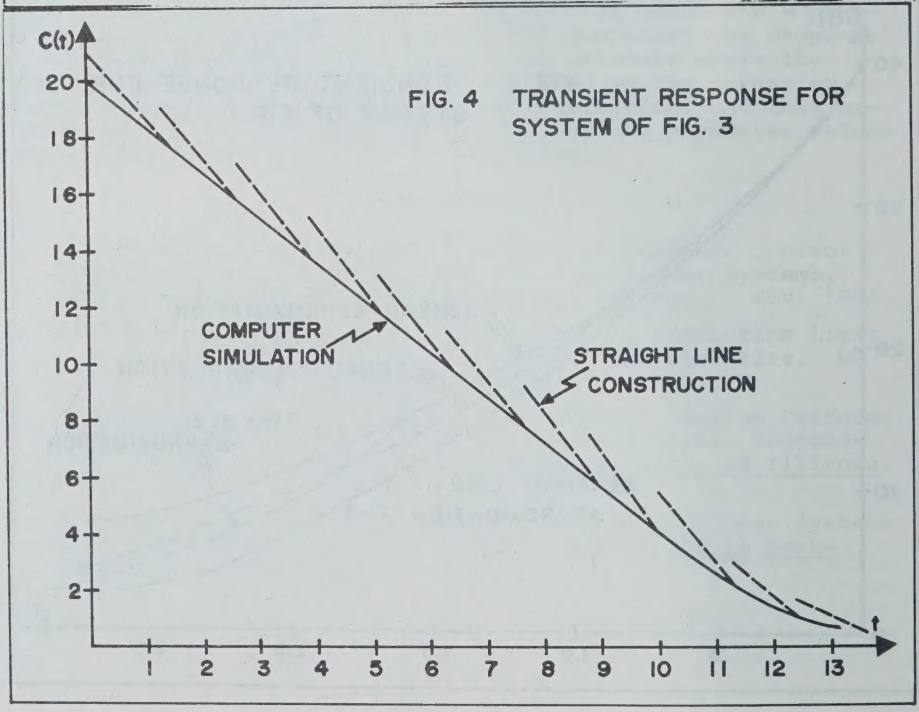


FIG. 4 TRANSIENT RESPONSE FOR SYSTEM OF FIG. 3

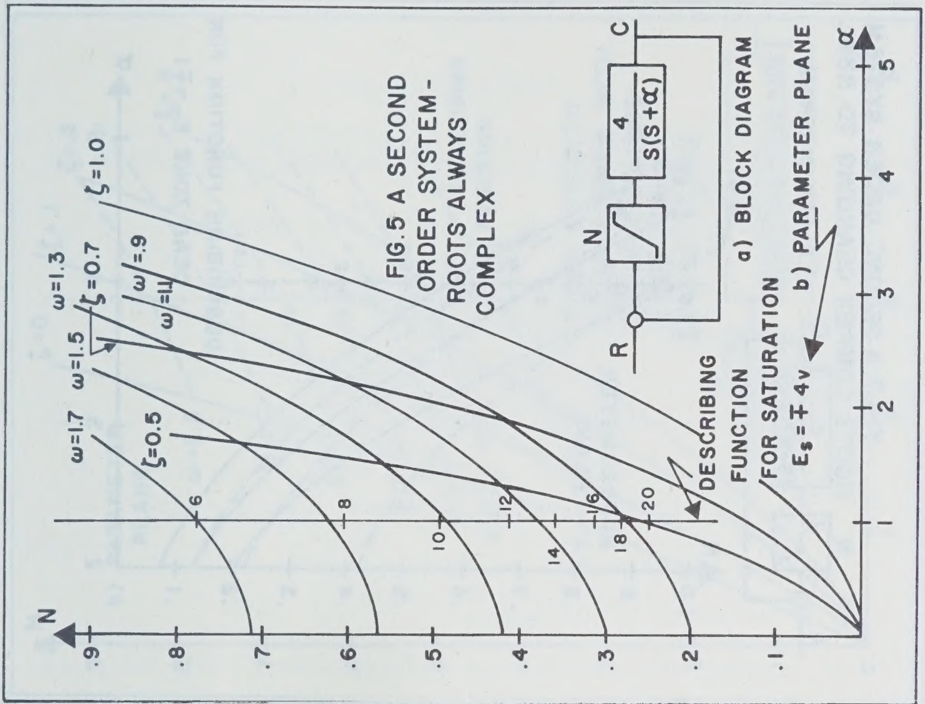


FIG.5 A SECOND ORDER SYSTEM- ROOTS ALWAYS COMPLEX

a) BLOCK DIAGRAM

b) PARAMETER PLANE

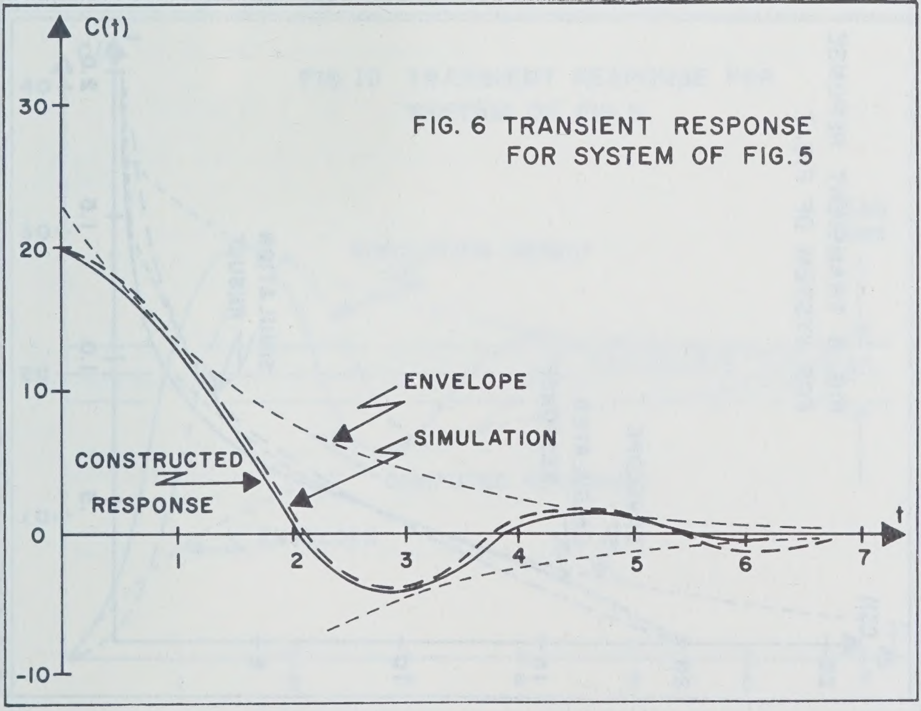


FIG. 6 TRANSIENT RESPONSE FOR SYSTEM OF FIG.5

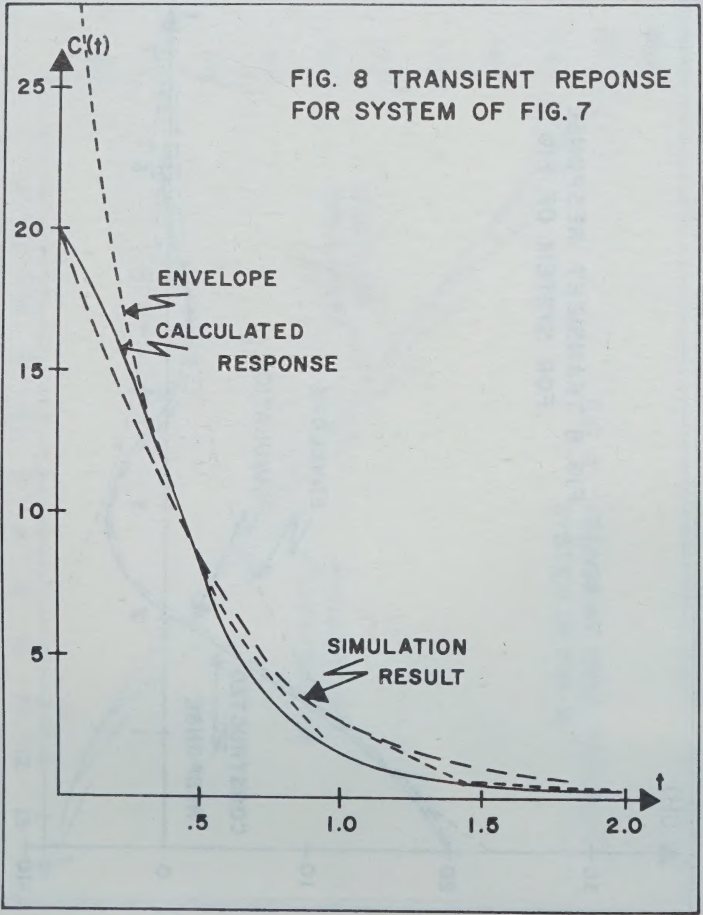


FIG. 8 TRANSIENT REPONSE FOR SYSTEM OF FIG. 7

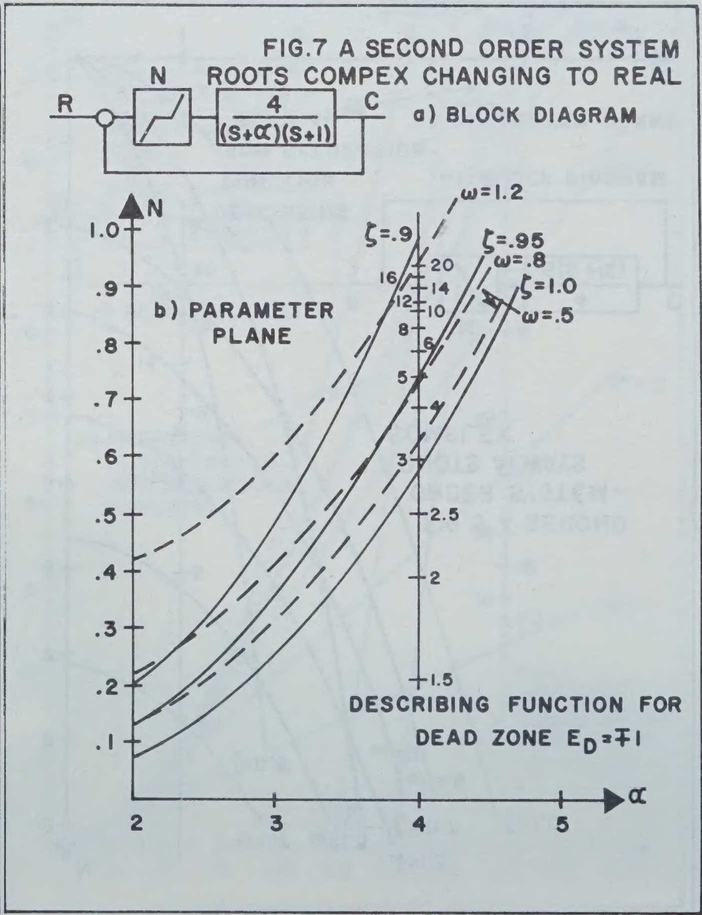


FIG.7 A SECOND ORDER SYSTEM ROOTS COMPEX CHANGING TO REAL

a) BLOCK DIAGRAM

b) PARAMETER PLANE

R
N
$\frac{4}{(S+\alpha)(S+1)}$
C
N
1.0
.9
.8
.7
.6
.5
.4
.3
.2
.1
ω=1.2
ζ=.9
ζ=.95
ω=.8
ζ=1.0
ω=.5
20
14
10
6
5
4
3
2.5
2
1.5
16
12
8
2
3
4
5
α
DESCRIBING FUNCTION FOR DEAD ZONE $E_D = \mp 1$

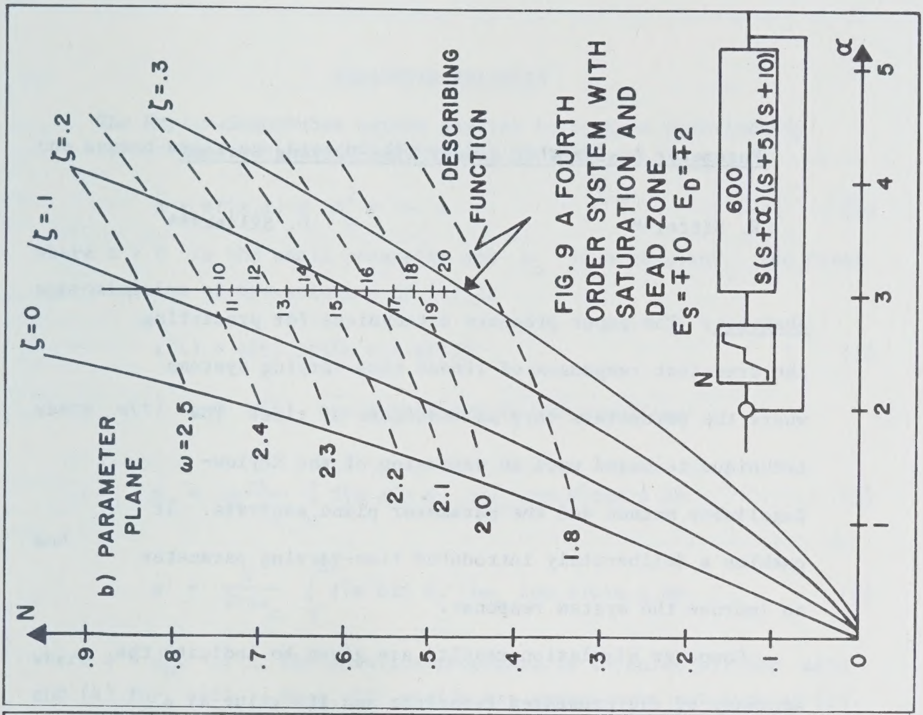
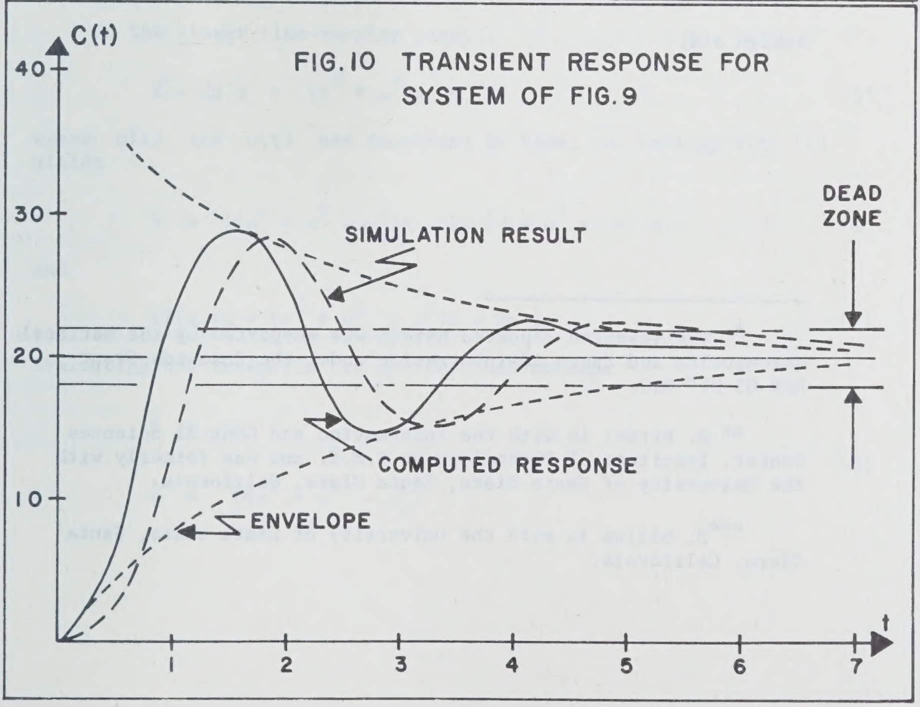


FIG. 9 A FOURTH ORDER SYSTEM WITH SATURATION AND DEAD ZONE $E_S = \mp 10 \quad E_D = \mp 2$

FIG. 10 TRANSIENT RESPONSE FOR SYSTEM OF FIG. 9



Parameter Analysis of Linear Time-Varying Systems*

R. Bittel** D. Siljak***

Abstract: The paper presents a technique for predicting the transient responses of linear time-varying systems where the parameters vary as functions of time. The technique is based upon an extension of the Krylov-Bogoliubov method and the parameter plane analysis. It enables a deliberately introduced time-varying parameter to improve the system response.

Computer simulation results are given to indicate the accuracy of the presented technique and its value as a design aid.

---

* The research reported herein was supported by the National Aeronautics and Space Administration under the Research Grant NGR-05-017-010.

** R. Bittel is with the Information and Control Sciences Center, Institute of Technology at S.M.U. and was formerly with the University of Santa Clara, Santa Clara, California.

*** D. Siljak is with the University of Santa Clara, Santa Clara, California.

PARAMETER ANALYSIS

The Krylov-Bogoliubov method applies to systems described by the second-order nonlinear differential equation

$$\ddot{x} + \varepsilon f(x,\dot{x}) + \omega_o^2 x = 0 \tag{1}$$

where $\varepsilon > 0$ is the small parameter and ω_o is a constant. The first approximation of the solution of (1) is

$$x(t) = a(t) \sin[\omega_o t + \phi(t)] \tag{2}$$

where $a(t)$ and $\phi(t)$ are determined by the equations

$$\dot{a} = \frac{-\varepsilon}{2\pi\,\omega_o} \int_o^{2\pi} f(a \sin\psi,\ a\omega_o \cos\psi)\cos\psi\, d\psi \tag{3}$$

and

$$\dot{\phi} = \frac{\varepsilon}{2\pi a\omega_o} \int_o^{2\pi} f(a \sin\psi,\ a\omega_o \cos\psi)\sin\psi\, d\psi \tag{4}$$

where $\psi = \omega_o t + \phi$. The solution procedure is to solve (3) for $a(t)$ and (4) for $\phi(t)$. Then, (2) yields the approximate solution of (1).

In the linear time-varying case,

$$\ddot{x} - 2\sigma\dot{x} + (\sigma^2 + \omega^2)x = 0 \tag{5}$$

where $\sigma(t)$ and $\omega(t)$ are functions of time; the analogy with (1) yields

$$\ddot{x} + [(\sigma^2 + \omega^2 - \omega_o^2)x - 2\sigma\dot{x}] + \omega_o^2 x = 0 \tag{6}$$

and

$$\varepsilon f(x,\dot{x}) = (\sigma^2 + \omega^2 - \omega_o^2)x - 2\sigma\dot{x} \tag{7}$$

Evaluating the integrals (3) and (4), one obtains

$$\begin{aligned} \dot{a} &= \sigma a \\ \dot{\phi} &= \tfrac{1}{2}(\sigma^2 + \omega^2 - \omega_o^2) \end{aligned} \tag{8}$$

After integration of (8), the approximate solution of (2) becomes

$$x(t) = a_o \exp\left(\int_o^t \sigma_1 dt\right) \sin\left(\int_o^t \omega_1 dt + \phi_o\right) \tag{9}$$

where a_o, ϕ_o are integration constants and σ_1, ω_1 stand for σ and $1/2(\sigma^2 + \omega^2 - \omega_o^2)$.

Differentiating (9) twice and substituting into (5), one obtains two equations

$$\begin{aligned} \dot{\sigma}_1 + \sigma_1^2 - 2\sigma_1\sigma - \omega_1^2 + \sigma^2 + \omega^2 &= 0 \\ 2\sigma_1\omega_1 + \dot{\omega}_1 - 2\omega_1\sigma &= 0 \end{aligned} \tag{10}$$

If σ and ω are slowly-varying functions of time, then $\dot{\sigma} \simeq \dot{\omega} \simeq 0$ and $\sigma_1 = \sigma$, $\omega_1 = \omega$ (see Appendix). In other words, the approximate solution of (5) is

$$x = a_o \exp\left(\int_o^t \sigma dt\right) \sin\left(\int_o^t \omega dt + \phi_o\right) \tag{11}$$

where $\sigma(t)$ and $\omega(t)$ are slowly-varying functions of time.

The above procedure can now be extended to high-order linear differential equations of the form

$$\sum_{k=o}^{n} a_k s^k x = 0, \quad s \equiv \frac{d}{dt} \tag{12}$$

where

$$a_k = a_k(\alpha,\beta), \quad (k = 0, 1, \ldots, n) \tag{13}$$

and $\alpha = \alpha(t)$, $\beta = \beta(t)$ are known functions of time.

The solution $x(t)$ of (12) can be assumed in the form (11) if the dominant pair of roots in the characteristic equation

$$F(s) \equiv \sum_{k=o}^{n} a_k s^k = 0 \tag{14}$$

can be determined by

$$s^2 - 2\sigma s + (\sigma^2 + \omega^2) = 0 \qquad (15)$$

where $\sigma = \sigma(t)$, $\omega = \omega(t)$ are slowly-varying in time. Note that the rate of change is not constrained in the variations of the parameters $\alpha(t)$ and $\beta(t)$, but rather in the changes of $\sigma(t)$ and $\omega(t)$. This allows a study of fast parameter variations provided the constraints on $\sigma(t)$ and $\omega(t)$ are not violated.

Once the functions $\alpha(t)$ and $\beta(t)$ are given, the parameter analysis [1-3] yields the functions $\sigma(t)$ and $\omega(t)$. Then, a tangential approximation (see the Appendix) can be used to reconstruct the solution $x(t)$ determined by equation (11). In majority of cases, however, the functions $\sigma(t)$ and $\omega(t)$ contain sufficient information about the solution $x(t)$ and it is not necessary to perform the construction of $x(t)$.

In a system design with computer simulation, the parameter variations may be chosen on the basis of the parameter plane analysis so that the functions $\sigma(t)$ and $\omega(t)$ result in a relatively fast response $x(t)$. This will be considered in the next section.

APPLICATION

Consider the system of Fig. 1 which has the characteristic equation

$$F(s) \equiv s^3 + (\alpha + 3)s^2 + (2\alpha + 2)s + \beta = 0 \qquad (16)$$

where $\alpha = k_1 k_{-1}$, $\beta = k_1 k_2$. The parameter $\alpha\beta$-plane is plotted in Fig. 2. For any point $M(\alpha,\beta)$ on the plane, all the characteristic roots can be evaluated without any additional calculations. In other words, one can find by inspection the region of dominancy of two complex conjugate roots $s = \sigma \pm j\omega$ and then determine the pair (σ,ω) as functions of the parameter values (α,β). In linear time-invariant case, this property of the parameter plane is used to choose the values of (α,β) so that the related root configuration results in a satisfactory system response. Specifically, for the points $M_o(0,6)$, $M_1(1,6)$ ..., $M_6(6,6)$ the associated responses are shown on Fig. 3.

Now, a time-varying parameter can be introduced into the system to improve the system response. The variation of the parameter is chosen so that it yields a lightly damped system for large values of the signal $x(t)$, and a highly damping system when the signal $x(t)$ becomes small. For various ramp functions $k_{-1}(t) = ct$, a set of signals are obtained as shown in Fig. 4. This indicated a considerable improvement when compared with Fig. 3. If the "best" ramp function $k_{-1} = 2.9\ t$ is chosen as indicated on Fig. 4, the responses for various initial conditions are shown on Fig. 5. There is no overshoot and the settling time is approximately 2 sec. irrespective to the initial conditions.

REFERENCES

1. Šiljak, D.D., "Analysis and Synthesis of Feedback Control Systems in the Parameter Plane, Part III - Nonlinear Systems," IEEE Transactions on Applications and Industry, vol. 83, No. 75, Nov. 1964, pp 466-473.

2. Šiljak, D.D., and M.R. Stojić, "Sensitivity Analysis of Self-Excited Nonlinear Oscillations," IEEE Transactions on Automatic Control, vol. AC-10, No. 4, Oct. 1965, pp 413-420.

3. Pellegrini, C.A., H.M. Yockey, and G.J. Thaler, "System Performance of Control Systems with Two Nonlinearities," Proceedings of the Fourth Allerton Conference on Circuit and System Theory, University of Illinois, Urbana, Ill., 1966, pp 877-886.

4. Huston, R.L., and L.F. Doty, "Note on the Krylov-Bogoliubov Method Applied to Linear Differential Equations," J. SIAM Appl. Math., vol. 14, No. 3, May, 1966, pp 424-428.

5. Bittel, R.H., "Parameter Analysis of Linear Time-Varying Systems," M.S. Thesis, University of Santa Clara, Santa Clara, California, 1967.

6. Popov, E.P., and I.P. Palitov, "Approximate Methods for Analyzing Nonlinear Automatic Systems," Moscow: State Press for Physics and Mathematical Literature, 1960 (in Russian). (English Translation, Foreign Tech. Div., AFSC, Wright-Patterson Air Force Base, Ohio, Rept. FTD-TT-62-910.)

7. Grensted, P.E.W., "Analysis of the Transient Response of Nonlinear Control Systems," Trans. of ASME, Vol. 80, Feb., 1958, pp 427-432.

8. Smol'nikov, L.P., and V.A. Khokhlov, "Designing Harmonically Linearized Systems with the Method of Averaged Roots," Avtomatika i Telemekhanika, vol. 27, No. 7, July, 1966, pp 86-91 (in Russian).

9. Šiljak, D.D., Nonlinear Systems: The Parameter Analysis and Design, John Wiley and Sons, New York, 1968.

APPENDIX

The assumption that equation

$$\ddot{x} - 2\sigma \dot{x} + (\sigma^2 + \omega^2)x = 0 \tag{5}$$

has the solution

$$x(t) = a_o \exp\left(\int_o^t \sigma_1 dt\right) \sin \left(\int_o^t \omega_1 dt + \phi_o\right) \tag{9}$$

led to the relationships

$$\begin{aligned} &\dot{\sigma}_1 + \omega_1^2 - 2\sigma_1\sigma - \omega_1^2 + \sigma^2 + \omega^2 = 0 \\ &2\sigma_1\omega_1 + \dot{\omega}_1 - 2\omega_1\sigma = 0 \end{aligned} \tag{10}$$

Solving (10) for σ_1 and ω_1^2, and neglecting higher order terms, one obtains

$$\begin{aligned} \sigma_1 &= \sigma - \frac{\dot{\omega}_1}{2\omega_1} \\ \omega_1^2 &= \dot{\sigma} + \omega^2 \end{aligned} \tag{17}$$

From (17), $2\omega_1\dot{\omega}_1 = \ddot{\sigma} + 2\omega\, \dot{\omega} \approx 2\omega\, \dot{\omega}$, and finally

$$\begin{aligned} \sigma_1 &= \sigma - \frac{\omega\dot{\omega}}{2(\dot{\sigma} + \omega^2)} \\ \omega_1^2 &= \dot{\sigma} + \omega^2 \end{aligned} \tag{18}$$

If σ and ω are slowly-varying functions of time, then $\dot{\sigma} \approx \dot{\omega} \approx 0$ and it may be concluded that as a first approximation one can consider $\sigma_1 = \sigma$, $\omega_1 = \omega$.

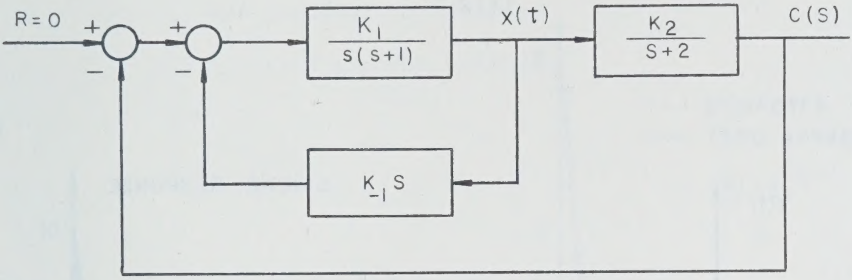
Now, since the functions $\sigma = \sigma(t)$ and $\omega = \omega(t)$ are known, it is possible to reconstruct the solution $x(t)$ by using tangential approximation. A variable time constant

$$T_i(t) = - \frac{1}{\sigma_i(t)} \tag{19}$$

is introduced to construct the amplitude variation $a(t)$. Read the value $\sigma_o = \sigma(o)$ from the diagram $\sigma(t)$ and calculate $T_o = T(o)$. Then, draw a straight line determined by the initial condition $x_o = x(o)$ and T_o as drawn in Fig. 6. A small portion of the amplitude envelope $a(t)$ is approximated by this line. Choosing a suitable time increment this straight line is terminated at $t = t_1$, and at t_1 a new value $a(t_1)$ is determined. For t_1, read $\sigma_1 = \sigma(t_1)$ and calculate $T_1 = T(t_1)$ from (19). The next segment of the envelope $a(t)$ from t_1 to t_2 is constructed and the process can be repeated from t_2 on. This yields the envelope $a(t)$.

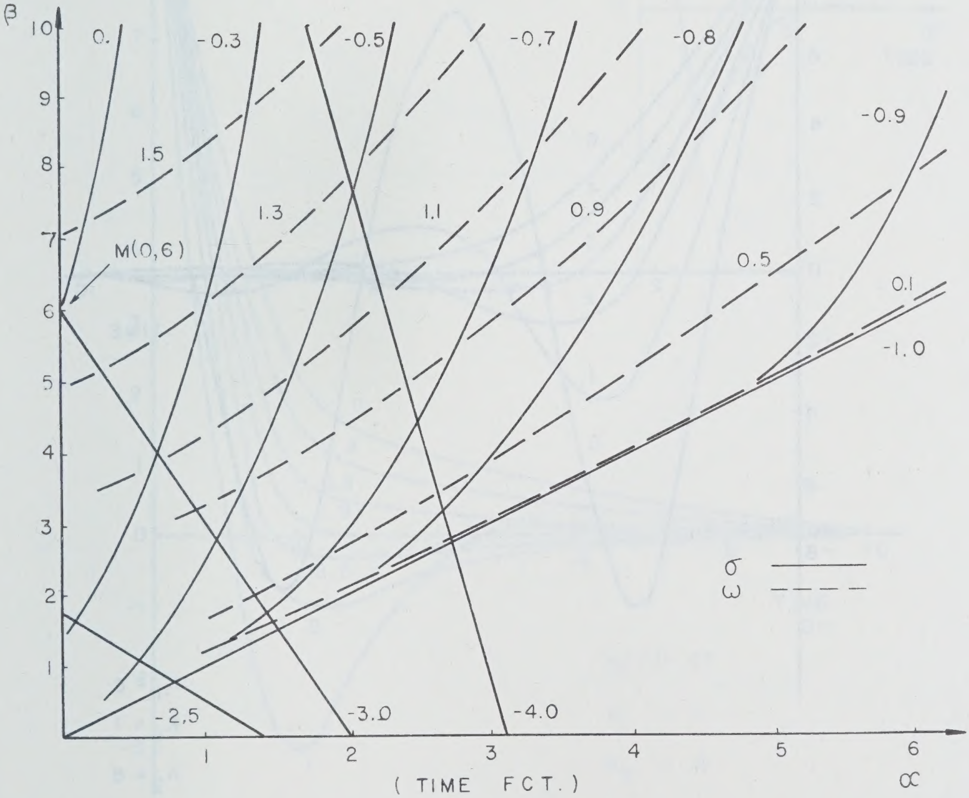
Since the phase is $\int_o^t \omega dt$, it can be determined from the area under the curve $\omega(t)$. Assuming $\phi_o = \pi/2$, the oscillations $x(t)$ will cross the time axis at π, 2π , etc., and have the extrema values at $3\pi/2$, $5\pi/2$, etc. Thus, the corresponding values of t are obtained from $\omega(t)$ by graphical integration (see the first diagram in Fig. 6).

Once the envelope $a(t)$, cross-over and extrema positions are found, the actual oscillation $x(t)$ can be reconstructed as shown on Fig. 6.



R = 0
+
−
+
−
$\frac{K_1}{s(s+1)}$
x(t)
$\frac{K_2}{S+2}$
C(S)
$K_{-1}S$

FIG. 1



β
10
9
8
7
6
5
4
3
2
1
0.
-0.3
-0.5
-0.7
-0.8
-0.9
1.5
1.3
1.1
0.9
0.5
0.1
-1.0
M(0,6)
-2.5
-3.0
-4.0
σ
ω
1
2
3
4
5
6
(TIME FCT.)
α

FIG. 2

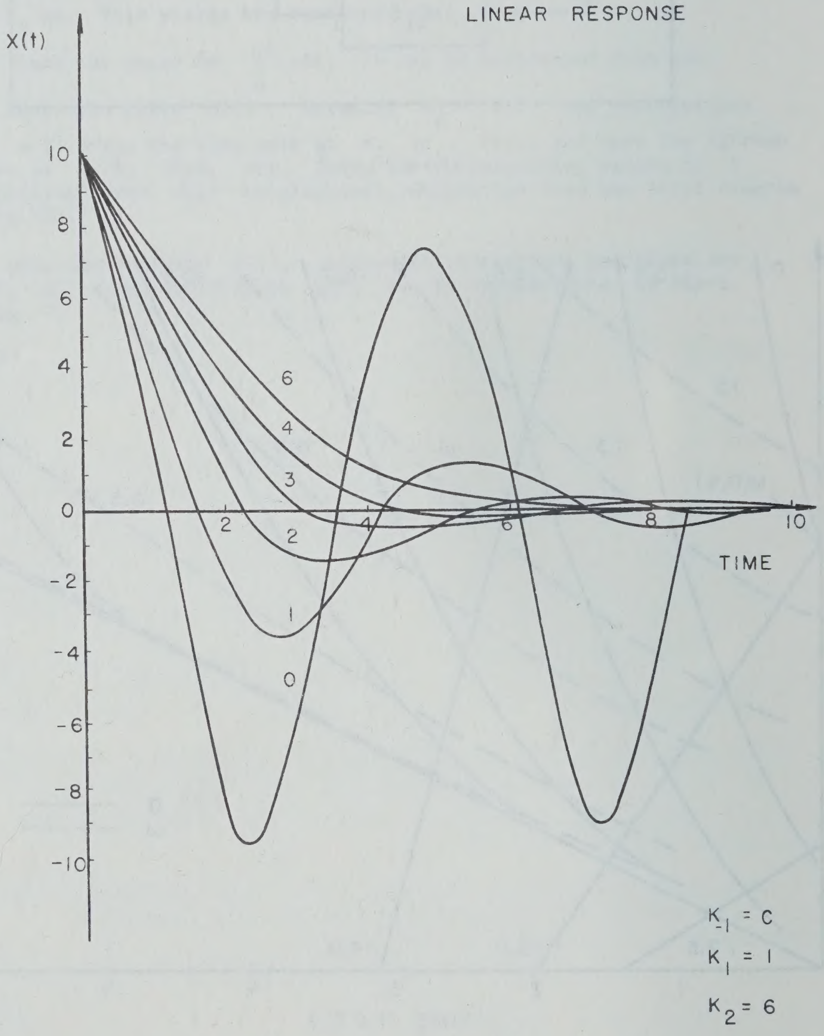

LINEAR RESPONSE
X(t)
10
8
6
4
2
0
-2
-4
-6
-8
-10
2
4
6
8
10
TIME
6
4
3
2
1
0
$K_{-1} = C$
$K_1 = 1$
$K_2 = 6$

FIG. 3

TIME – VARYING RESPONSE

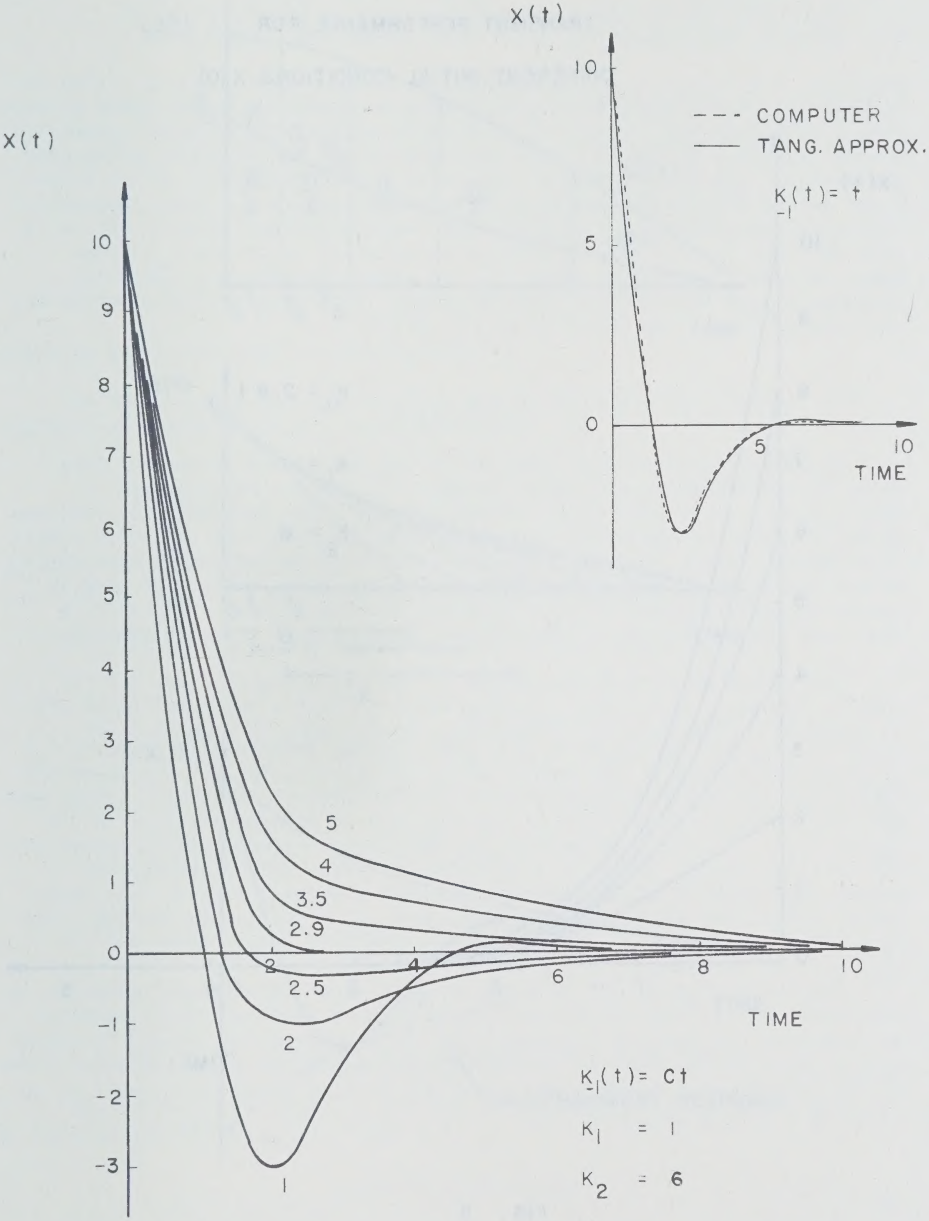
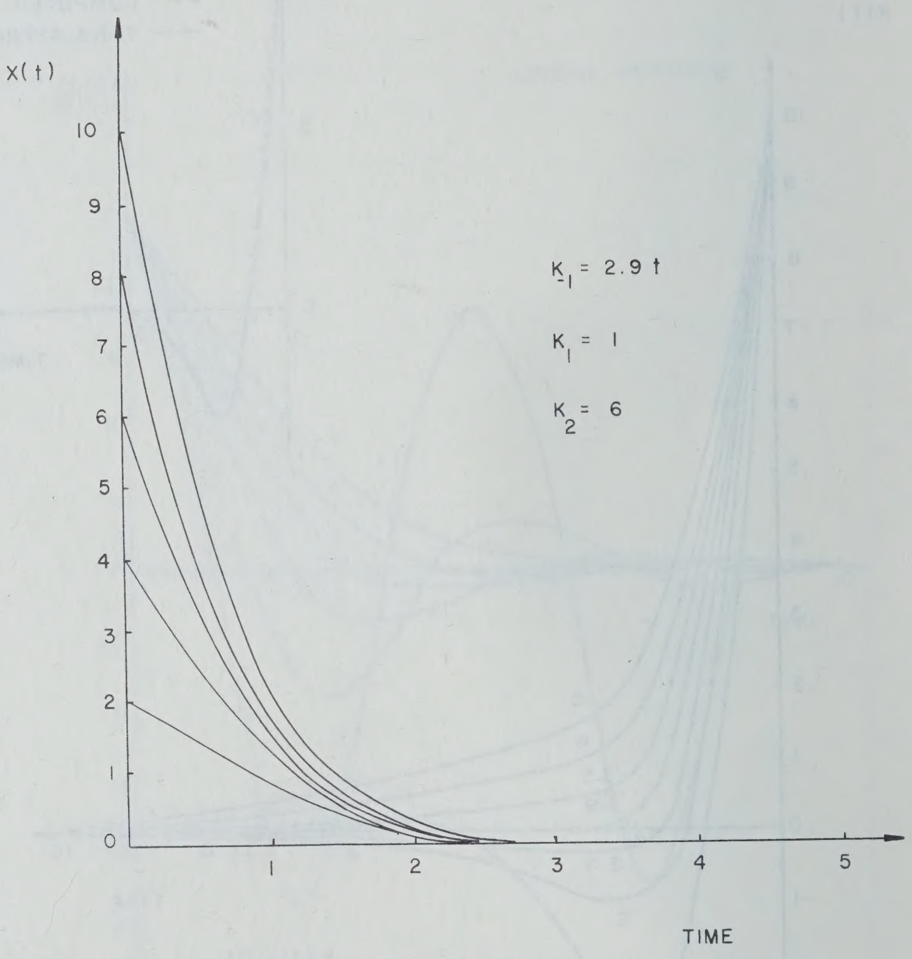


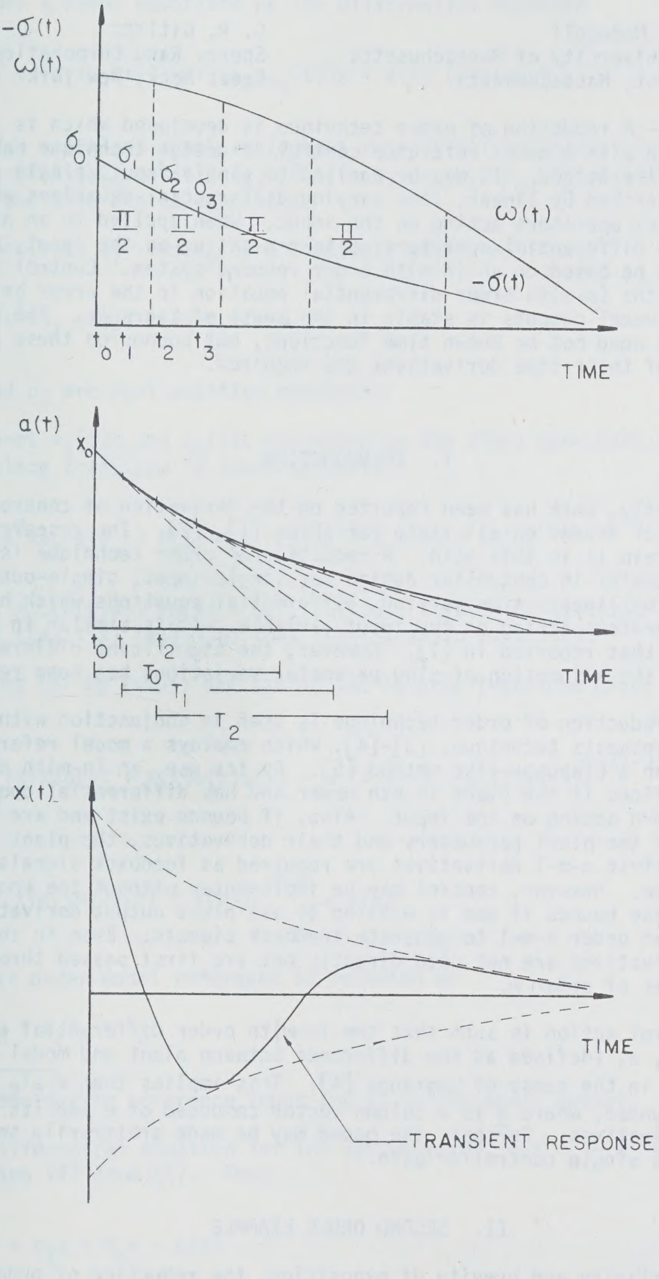
FIG. 4



TRANSIENT PERFORMANCE FOR

DIFFERENT INITIAL CONDITIONS X(0)

FIG. 5



$-\sigma(t)$
$\omega(t)$
σ_0
σ_1
σ_2
σ_3
$\frac{\pi}{2}$
$\frac{\pi}{2}$
$\frac{\pi}{2}$
$\frac{\pi}{2}$
$\omega(t)$
$-\sigma(t)$
t_0
t_1
t_2
t_3
TIME
$a(t)$
x_0
t_0
t_1
t_2
T_0
T_1
T_2
TIME
$X(t)$
TIME
TRANSIENT RESPONSE

FIG . 6

CONTROL OF LINEAR TIME VARYING PLANTS WITH DIFFERENTIAL OPERATORS ACTING ON THE INPUT

R. V. Monopoli
The University of Massachusetts
Amherst, Massachusetts

C. R. Gitlitz
Sperry Rand Corporation
Great Neck, New York

Abstract -- A reduction of order technique is developed which is useful in conjunction with a model reference controller design technique based on a Liapunov-like method. It may be applied to single-input, single-output plants described by linear, time varying differential equations which have differential operators acting on the input. When applied to an nth order plant with differential operators up to m<n acting on the input, controller design may be based on an (n-m)th order reduced system. Control action is such that the (n-m)th order differential equation in the error between plant and model outputs is stable in the sense of Lagrange. The plant parameters need not be known time functions, but bounds on these parameters and some of their time derivatives are required.

I. INTRODUCTION

Recently, work has been reported on the derivation of control laws which do not depend on all state variables [1], [2]. The research reported herein is in this vein. A reduction of order technique is developed which is useful in controller design for single-input, single-output plants described by linear, time varying, differential equations which have differential operators acting on the input variable. It is similar in some respects to that reported in [1]. However, the significant difference is that here the assumption of slow parameter variations has been removed.

The reduction of order technique is used in conjunction with a controller synthesis technique, [3]-[4], which employs a model reference and is based on a Liapunov-like method [5]. By its use, an (n-m)th order model suffices if the plant is nth order and has differential operators up to order m<n acting on the input. Also, if bounds exist and are known for certain of the plant parameters and their derivatives, the plant output and only its first n-m-1 derivatives are required as feedback signals in the control law. However, control may be implemented without the knowledge of all of these bounds if one is willing to use plant output derivatives higher than order n-m-1 to generate feedback signals. Even in this case, these derivatives are not used directly but are first passed through a low pass filter of order m.

Control action is such that the (n-m)th order differential equation in the error, e, (defined as the difference between plant and model outputs) is stable in the sense of Lagrange [4]. This implies that $\sqrt{\underline{e}^T\underline{e}}$ is ultimately bounded, where $\underline{e}$ is a column vector composed of e and its first n-m-1 derivatives. Further, the bound may be made arbitrarily small by increasing a single controller gain.

II. SECOND ORDER EXAMPLE

For clarity and brevity of exposition, the reduction of order technique is developed by considering in detail its application to a second order plant only. The extension to higher order systems is outlined briefly in the next section.

Consider a plant described by the differential equation

$$\ddot{x} + (a_1+\alpha_1(t))\,\dot{x} + (a_o+\alpha_o(t))x = K(t)\ (\dot{u}+b(t)u) \qquad (1)$$

where u and x are plant input and output respectively; $K(t)>\varepsilon>0, |\alpha_1(t)|\le M_1$, $|\alpha_o(t)|<M_o$, $|K(t)|\le C_o$, $|b(t)|\le B_o$, $|\dot{K}(t)|\le C_1$; ε, M_1, M_o, C_o, C_1, and B_o are are known, positive finite constants.

The constants a_1 and a_o are chosen in such a way that

$$(s^2+a_1\ s+a_o) = (s+p_1)(s+p_2) \qquad (2)$$

where p_1 and p_2 are real positive constants.

The terms $\alpha_1(t)\dot{x}$ and $\alpha_o(t)x$ are moved to the right hand side of (1) and the Laplace transform is taken to yield

$$(s^2+a_1\ s+a_o s)\ X(s) = \sum_{i=1}^{4} F_i(s) + a_1 x(o) + s x(o) \qquad (3)$$

where $F_1(s) = L\{K(t)\dot{u}\}$, $F_2(s) = L\{K(t)b(t)u\}$, $F_3(s) = L\{-\alpha_1(t)\dot{x}\}$, and and $F_4(s) = L\{\alpha_o(t)x\}$ (assuming that these transforms exist).

Dividing (3) by $(s+p_1)$ and taking the inverse transform gives

$$\dot{x} + p\ x = \int_o^t \{K(\lambda)\dot{u}(\lambda) + K(\lambda)b(\lambda)u(\lambda) - \alpha_1(\lambda)\dot{x}(\lambda) - \alpha_o(\lambda)x(\lambda)\}\varepsilon^{-p_1(t-\lambda)}d\lambda$$

$$+ \{(a_1-p_1)x(o) + \dot{x}(o)\}\varepsilon^{-p_1 t} + x(o) \qquad (4)$$

A first order model reference is selected as

$$\dot{x}_d + p_2 x_d = K_o r \qquad (5)$$

where r is the system reference input and x_d is the model output.

The differential equation for the system error, $e = x_d - x$ is found by subtracting (4) from (5). Thus

$$\dot{e} + p_2 e = K_o r - g(t) \qquad (6)$$

where g(t) is the right hand side of (4).

The integrals of g(t) involving $\dot{u}$ and $\dot{x}$ may be integrated by parts to give

$$\int_o^t K(\lambda)\dot{u}(\lambda)\varepsilon^{-p_1(t-\lambda)}\,d\lambda = K(t)u(t) - K(o)u(o)\varepsilon^{-p_1 t}$$

$$-\int_o^t u(\lambda)(p_1 K(\lambda)+\dot{K}(\lambda))\varepsilon^{-p_1(t-\lambda)}\,d\lambda \tag{7a}$$

$$\int_o^t \alpha_1(\lambda)\dot{x}^{-p_1(t-\lambda)}\,d\lambda = \alpha_1(t)x(t) - \alpha_1(o)x(o)\varepsilon^{-p_1 t}$$

$$-\int_o^t x(\lambda)(p_1\alpha_1(\lambda)+\dot{\alpha}_1(\lambda))\varepsilon^{-p_1(t-\lambda)}\,d\lambda \tag{7b}$$

It is to be noted that the integration by parts performed in (7) eliminated $\dot{u}$ and $\dot{x}$ in (6), but introduced $\dot{K}(t)$ and $\dot{\alpha}_1(t)$. The exchange of $\dot{x}$ for $\dot{\alpha}_1(t)$ may be advantageous in generating the control signal. This point is considered further below.

The Liapunov-like approach to the controller design is introduced by associating a positive definite function of e with (6). Let this function be

$$V(e) = e^2/2 \tag{8}$$

for which the time derivative is

$$\dot{V} = e\dot{e} \tag{9}$$

With (7a) and (7b) substituted into (6), it can be seen that the control signal given by (10) below assures that (6) is stable in the sense of Lagrange [5],

$$u = \{K_1 w + K_2|r| + K_3|x| + K_4 z + K_5|x(o)|\} \text{ sat } ke \tag{10}$$

where the following definitions pertain:

$$\text{sat } ke = \begin{cases} +1 \text{ if } e > 1/k \\ ke \text{ if } -1/k \leq e \leq 1/k \\ -1 \text{ if } e < -1/k \end{cases} \text{ where } k \text{ is an arbitrary positive constant}$$

$$\dot{w} + p_1 w = |u|$$

$$\dot{z} + p_1 z = |x|$$

$$K_1 \geq (\max_t |K(t)(b(t)-p_1) - \dot{K}(t)|)\,(\max_t (1/K(t)))$$

$$K_2 \geq \max_t |K_o/K(t)|$$

$$K_3 \geq \max_t |\alpha_1(t)/K(t)|$$

$$K_4 \geq \left(\max_t |p_1\alpha_1(t)+\dot{\alpha}_1(t)-\alpha_o(t)|\right)\left(\max_t (1/K(t))\right)$$

$$K_5 \geq \max_t (1/K(t))$$

In deriving (10), inequalities of the following form were employed:

$$\frac{1}{K(t)} \int_o^t (p_1 K(\lambda)+\dot{K}(\lambda)-K(\lambda)b(\lambda))u(\lambda)\varepsilon^{-p_1(t-\lambda)}\, d\lambda$$

$$\leq \frac{1}{K(t)} \int_o^t |p_1 K(\lambda)+\dot{K}(\lambda)-K(\lambda)b(\lambda)|\; |u(\lambda)|\varepsilon^{-p_1(t-\lambda)}\, d\lambda \leq K_1|w|$$

The gains K_1 through K_5 may be computed provided $K(t)$, $\alpha_1(t)$, $\alpha_o(t)$, $b(t)$, and $\dot{K}(t)$ satisfy the requirements specified in (1) and also provided an M_{11} ($|\dot{\alpha}_1(t)| \leq M_{11}$) exists and is known. An upper bound for $\dot{K}(t)$ must always be known to implement this control. However, if an upper bound for $|\dot{\alpha}_1(t)|$ is not known, control may still be effected if one is willing to use a filtered form of $\dot{x}$ to generate u. An advantage is gained if the bound M_{11} is known, i.e. $\dot{x}$ is not required to implement (10).

Consider the case when an upper bound for $|\dot{\alpha}_1(t)|$ is not known. Then, instead of employing (7b), the integral involving $\dot{x}$ is retained in (6). In this case the control signal is

$$u = \{K_1 w + K_2|r| + K_4^1 z + K_5|x(o)| + K_6 v\} \text{ sat } ke \tag{11}$$

where

$$\dot{v} + p_1 v = |\dot{x}|$$

$$K_4^1 = \left(\max_t |\alpha_o(t)|\right)\left(\max_t (1/K(t))\right)$$

$$K_6 = \left(\max_t |\alpha_1(t)|\right)\left(\max_t (1/K(t))\right)$$

Although $\dot{x}$ is required to generate (11), it is first passed through a low pass filter with time constant p_1^{-1} which will help in eliminating high frequency noise.

III. APPLICATION TO HIGHER ORDER SYSTEMS

This control technique employing reduction of order can be applied to plants of any order described by equations of the form

$$x^n + \sum_{i=0}^{n-1} (a_i+\alpha_i(t))x^i = K(t)(u + \sum_{j=0}^{m-1} b_j(t)u^j) \qquad (12)$$

where superscripts denote time derivatives; $K(t)>\varepsilon>o$, $|\alpha_i(t)| \leq M_i$ for $i = 0, 1,2,\dots n-1$, $|K^i(t)| \leq C_i$ for $i = 0, 1,\dots m$, $| b_j(t)|< B_j$ for $j = 0,1,\dots m-1$, $|b_j^i(t)| \leq B_{ji}$ for $j = 1,2,\dots m-1$; ε, M_i, C_i, B_j, and B_{ji} are known positive, finite constants.

Though the extension to higher order systems is complicated in its details, it is relatively straight forward conceptually. Because of the complications in details, it is worthwhile commenting on several points relating to the extension.

For an nth order plant with $n-m = q$, the constants $a_0, a_1,\dots a_{n-1}$ must be chosen so that

$$s^n + a_{n-1} s^{n-1} +\dots a_o = (\prod_{i=1}^{m} (s+p_i))(s^q+c_{q-1} s^{q-1}+\dots c_o) \qquad (13)$$

where $p_i >0$ for $i = 1,2,\dots m$, and the qth order polynomial has roots with negative real parts. In the steps corresponding to (3) and (4) of the second order example, the transformed plant equation is divided by $\prod_{i=1}^{m}(s+p_i)$. Thus, the order of the reduced plant equation is q, and the qth order model is specified by the transfer function $K_o(s^q+ c_{q-1}s^{q-1}+..c_o)^{-1}$.

The positive definite function to be employed in the general case is $V(\underline{e}) = \underline{e}^T P\underline{e}$ where $\underline{e}^T = (e, \dot{e},\dots e^{q-1})$, and P is a positive definite q x q symmetric matrix. P is the unique solution of $-Q = A^TP +PA$, where Q is positive definite and A is the q x q system matrix which results when the error differential equation is written in phase variable form. The control signal is then developed in the same way as (10). Details may be found in [3] or [4].

A typical term on the right hand side of the reduced plant equation will be a convolution integral, e.g.

$$\int_o^t \alpha_{n-1}(\lambda)x^{n-1}(\lambda)h(t-\lambda)d\lambda \qquad (14)$$

where $h(t) = L^{-1}\{\prod_{i=1}^{m} (s+p_i)^{-1}\}$

For example, if n = 3, and m = 2, then this is

$$\int_0^t \alpha(\lambda)\,\ddot{x}(\lambda)\,(1/(p_1-p_2))(-\varepsilon^{-p_1(t-\lambda)}+\varepsilon^{-p_2(t-\lambda)})\,d\lambda \qquad (15)$$

If bounds for $|\dot{\alpha}_2(t)|$ and $|\ddot{\alpha}_2(t)|$ exist and are known, (15) may be integrated twice by parts (similar to the procedure described in connection with (7b)) leading to terms in u which do not involve $\ddot{x}$, but x instead. In the event these bounds do not exist or are not known, the appropriate term in u may still be generated by feeding $|\ddot{x}|$ through a filter with transfer function $[(s+p_1)(s+p_2)]^{-1}$.

IV. CONCLUSIONS

Single-input, single-output plants described by nth order linear time varying differential equations with differential operators up to order m<n acting on the input variable may be made to behave like an (n-m)th order model reference with the first n-m-1 output derivatives employed as feedback signals in the control law. This result obtains if upper and lower bounds exist and are known for certain of the plant parameters and their time derivatives. Control may be implemented without knowledge of all of these bounds, but in this case plant output derivatives higher than order n-m-1 must be employed to generate feedback signals after first being passed through a low pass filter of order m.

REFERENCES

1. R. V. Monopoli, "Control of Linear Plants with Zeros and Slowly Varying Parameters," I.E.E.E. Trans. on Auto. Control, Vol. AC-12. No. 1, Feb. 1967, pp. 80-83.

2. Z. V. Rekasius, "Optimal Linear Regulators with Incomplete State Feedback," I.E.E.E. Trans. on Auto. Control, Vol. AC-12, No. 3, June 1967, pp. 296-299.

3. L. P. Grayson, "Design Via Liapunov's Second Method," Preprints of 1963 JACC, New York; A.I.C.E., June 1963.

4. R. V. Monopoli, "Engineering Aspects of Control System Design Via the Direct Method of Liapunov," NASA Report CR-654, Washington, D. C., December, 1966.

5. J. LaSalle and S. Lefschetz, Stability by Liapunov's Direct Method with Applications, Academic Press, New York, N. Y., 1961.

THE DESIGN OF SUBOPTIMAL LINEAR TIME-VARYING SYSTEMS

DAVID L. KLEINMAN
Research Engineer
Bolt, Beranek and Newman
Cambridge, Mass.

MICHAEL ATHANS
Associate Professor
Department of Electrical Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts

ABSTRACT

This paper deals with the determination of suboptimal feedback laws for the control of linear time-varying systems with quadratic performance criteria. Easily implementable time-functions are used to generate the required time-varying gains; free constant parameters in the control law description are chosen so as to minimize an "averaged" control cost. A simple example is included to illustrate the theory.

INTRODUCTION

One of the most widely studied problems in the field of optimal control is the optimal linear regulator problem,[1-3] that is, the control of linear systems with respect to quadratic performance criteria. The problem's importance and application to wide segments of automatic control, such as optimal tracking problems as well as linear filtering and prediction problems,[4] has prompted its extensive study in recent years.

It is well-known that the optimal control for this problem, $\underline{u}^*(t)$, for $t\epsilon[t_0,t_1]$, is generated by a linear feedback control law of the form

$$\underline{u}^*(t) = - \underline{L}^*(t)\underline{x}(t)$$

where $\underline{x}(t)$ is the current state of the system and $\underline{L}^*(t)$ is a matrix of feedback gains. The elements of the matrix $\underline{L}^*(t)$ are computed from the solution of a nonlinear matrix differential equation (the Riccati equation) which lies at the heart of the optimization problem.

In spite of its apparent mathematical simplicity, there are certain engineering shortcomings associated with the implementation of the time-varying feedback gains. Because of the computational instability of the Riccati equation solution in the forward time direction,[2,3] it is not possible to accurately compute $\underline{L}^*(t)$, $t>t_0$, in an on-line manner.* It therefore becomes necessary to first solve the Riccati equation off-line, for $t<t_1$, by starting at $t=t_1$ with an appropriate boundary condition. One then stores $\underline{L}^*(t)$ on tape, or in the computer memory of the feedback controller, to be "played back" upon command in real time. This method of implementing the optimal control is difficult and often impractical for multi-input, multi-output systems due to the complex circuitry requirements for synchronous playback of many time-varying signals, or due to the excessive computer-memory requirements.

In this paper a suboptimal control scheme for linear regulator systems is proposed. The goal is to determine a linear feedback law which is relatively easy to implement, yet one which results in near-optimal system performance. The method of approach trades mathematical optimality in return for practical usefulness by prespecifying the structural form of the time-varying feedback gains, while leaving various free parameters to be chosen in an optimal fashion. In this manner, a suboptimal linear regulator problem is precisely formulated and necessary conditions for its solution are derived. It is shown that under suitable assumptions, the suboptimal control can approximate the true optimal control arbitrarily closely. Lastly, a suboptimal control law is calculated for a second-order example.

---

*This is not the case in the filtering problem for which the Riccati equation solution is stable at $t\to\infty$. Nonetheless, the theoretical results of this paper remain applicable to linear filtering problems for which real-time computational facility is required.

PRELIMINARY CONSIDERATIONS

In this section the optimal linear regulator problem is formulated and the salient features of the optimal solution are presented. The concept of a cost-matrix for linear regulator problems is introduced and relationships are derived between different cost-matrices.

Consider the linear dynamical system characterized by

$$\frac{d}{dt}\underline{x}(t) = \underline{A}(t)\underline{x}(t) + \underline{B}(t)\underline{u}(t) \qquad (1)$$

where $\underline{x}(t)$ is the n-dimensional state vector, $\underline{u}(t)$ is the r-dimensional unconstrained control input; the nxn and nxr matrices $\underline{A}(t)$ and $\underline{B}(t)$ are locally integrable. Suppose that t_0 and $t_1 > t_0$ are fixed initial and final times, respectively, and that

$$\underline{x}_0 \overset{\Delta}{=} \underline{x}(t_0) \qquad (2)$$

is a given initial state for Eq. 1.

If $\underline{u}(.)$ is a given control input over the prespecified time-interval $[t_0,t_1]$, let $\underline{x}_u(t) = \phi(t;t_0,\underline{x}_0,\underline{u}(.))$ denote the solution of the system Eq. 1 starting from $\underline{x}_0$ at time t_0 and generated by the control $\underline{u}(.)$. The optimal linear regulator problem is to determine the control $\underline{u}(.)$ on $[t_0,t_1]$ which minimizes the quadratic cost functional:

$$J(\underline{x}_0,t_0;\underline{u}(.)) = \underline{x}_u'(t_1)\underline{F}\underline{x}_u(t_1) + \int_{t_0}^{t_1}[\underline{x}_u'(t)\underline{Q}(t)\underline{x}_u(t) + \underline{u}'(t)\underline{R}(t)\underline{u}(t)]dt \qquad (3)$$

where the terminal state $\underline{x}_u(t_1)$ is unconstrained, $\underline{F}$ is a positive semi-definite matrix, while $\underline{Q}(t) \geqslant \underline{0}$, $\underline{R}(t) > \underline{0}$, and are both continuous in t.

As is well-known,[1-3] the optimal control $\underline{u}^*(.)$ is generated by the linear feedback law

$$\underline{u}^*(\underline{x}(t),t) = -\underline{R}^{-1}(t)\underline{B}'(t)\underline{K}(t)\underline{x}(t) \overset{\Delta}{=} -\underline{L}^*(t)\underline{x}(t) \qquad (4)$$

where $\underline{K}(t)$ is the unique (positive semidefinite) solution[5] of the matrix Riccati differential equation

$$\dot{\underline{K}}(t) = -\underline{A}'(t)\underline{K}(t) - \underline{K}(t)\underline{A}(t) - \underline{Q}(t) + \underline{K}(t)\underline{B}(t)\underline{R}^{-1}(t)\underline{B}'(t)K(t) \qquad (5)$$

satisfying the boundary condition at the terminal time t_1,

$$\underline{K}(t_1) = \underline{F} \qquad (6)$$

Furthermore, the matrix $\underline{K}(t)$ has the property that for arbitrary $t \varepsilon (-\infty,t_1)$ and arbitrary $\underline{x} \varepsilon R_n$,

$$J(\underline{x},t;\underline{u}(.))\Big|_{\underline{u}=-\underline{L}^*(t)\underline{x}(t)} = \min_{\underline{u}(.)} J(x,t;\underline{u}(.)) = \underline{x}'\underline{K}(t)\underline{x} \qquad (7)$$

In particular, substituting $\underline{x}_0$ and t_0 for $\underline{x}$ and t, respectively, Eq. 7 yields

$$\min_{\underline{u}(.)} J(\underline{x}_0,t_0;\underline{u}(.)) = \underline{x}_0'\underline{K}(t_0)\underline{x}_0 \qquad (8)$$

Consider, now, the control law $\underline{u}_L(\underline{x}(t),t) = -\underline{L}(t)\underline{x}(t)$ with $\underline{L}(t) \neq \underline{L}^*(t)$. Clearly, $\underline{u}_L(.)$ is not optimal. Therefore, by comparing the costs associated with the use of $\underline{L}(t)$ as opposed to $\underline{L}^*(t)$, one can obtain bounds on the matrix* $\underline{K}(t)$. This is illustrated by the following lemma, together with its constructive proof.

Lemma I : Let $\underline{L}(t)$ *be an arbitrary* rxn *time-varying matrix defined for* $t\varepsilon[t_0,t_1]$. *Let* $\underline{V}_L(t)$ *denote the (unique positive semidefinite) solution of the linear matrix differential equation*

$$\underline{\dot{V}}(t)=-\underline{V}(t)[\underline{A}(t)-\underline{B}(t)\underline{L}(t)]-[\underline{A}(t)-\underline{B}(t)\underline{L}(t)]'\underline{V}(t)-\underline{Q}(t)-\underline{L}'(t)\underline{R}(t)\underline{L}(t) \qquad (9)$$

satisfying the boundary condition $\underline{V}(t_1) = \underline{F}$. *Then*

$$\underline{K}(t) \leqslant \underline{V}_L(t) \quad \underline{\text{for all}} \; t\varepsilon[t_0,t_1] \qquad (10)$$

Proof : Apply the control input $u_L(\underline{x}(t),t)=-\underline{L}(t)\underline{x}(t)$ to the system (1). The resulting system behavior is then governed by the "closed-loop" equation

$$\underline{\dot{x}}(t) = [\underline{A}(t)-\underline{B}(t)\underline{L}(t)]\underline{x}(t) \qquad (11)$$

Let $\underline{\Phi}_L(t,t_0)$ denote the transition matrix corresponding to $[\underline{A}(t)-\underline{B}(t)\underline{L}(t)]$, i.e., $\underline{\Phi}_L(t,t_0)$ satisfies the matrix differential equation

$$\frac{d}{dt}\underline{\Phi}_L(t,t_0) = [\underline{A}(t)-\underline{B}(t)\underline{L}(t)]\underline{\Phi}_L(t,t_0);\underline{\Phi}_L(t_0,t_0)=\underline{I} \qquad (12)$$

If t and $\underline{x}$ are arbitrary elements of (t_0,t_1) and R_n, respectively, the cost resulting from application of $\underline{u}_L(.)$ is found to be

$$J(\underline{x},t;\underline{u}_L(.)) = \underline{x}'\underline{V}(t)\underline{x} \qquad (13)$$

where

$$\underline{V}(t)\triangleq\underline{\Phi}_L'(t_1,t)\underline{F}\underline{\Phi}_L(t_1,t)+\int_t^{t_1}\underline{\Phi}_L'(\tau,t)[\underline{Q}(\tau)+\underline{L}'(\tau)\underline{R}(\tau)\underline{L}(\tau)]\underline{\Phi}_L(\tau,t)d\tau \qquad (14)$$

Upon differentiating both sides of Eq. 14 with respect to t one obtains Eq. 9 with the boundary condition $\underline{V}(t_1) = \underline{F}$. Clearly, Eq. 14 is the unique (positive semidefinite) solution of Eq. 9. Let $\underline{V}_L(t)$ denote this solution, emphasizing its dependence upon $\underline{L}(t)$. Finally, since $\underline{u}_L(.)$ is not the optimal control, one has

$$\underline{x}'\underline{K}(t)\underline{x} = J(\underline{x},t;\underline{u}^*(.)) \leqslant J(\underline{x},t;\underline{u}_L(.)) = \underline{x}'\underline{V}_L(t)\underline{x}$$

Hence, since $\underline{x}$ is arbitrary, the conclusion (10) is reached. Q.E.D.

Inasmuch as

$$J(\underline{x},t;\underline{u}(.))\Big|_{\underline{u}=-\underline{L}(t)\underline{x}(t)} = \underline{x}'\underline{V}_L(t)\underline{x} \qquad (15)$$

we shall henceforth *call the matrix* $\underline{V}_L(t)$ *given by Eq. 14, and satisfying Eq. 9, the cost matrix associated with* $\underline{L}(t)$. Note that according to this definition the cost matrix associated with $\underline{L}^*(t)$ is $\underline{K}(t)$.

In Appendix A, several expressions for the difference between cost matrices associated with different $\underline{L}(t)$ are derived. An immediate application of these results yields a simple expression for $\underline{V}_L(t)-\underline{K}(t)$. Substituting $\underline{L}(t)$ for $\underline{L}_1(t)$ and $\underline{L}^*(t)$ for $\underline{L}_2(t)$ in Eq. A.9 one obtains, since $\underline{L}^*(t)=\underline{R}^{-1}(t)\underline{B}'(t)\underline{V}_2(t)$, that

$$\underline{V}_L(t)-\underline{K}(t)= \int_t^{t_1}\underline{\Phi}_L'(\tau,t)[\underline{L}(\tau)-\underline{L}^*(\tau)]'\underline{R}(\tau)[\underline{L}(\tau)-\underline{L}^*(\tau)]\underline{\Phi}_L(\tau,t)d\tau \qquad (16)$$

---

*If $\underline{X}$ and $\underline{Y}$ are positive semidefinite matrices, the notation $\underline{X}>\underline{Y}[\underline{X}\geqslant\underline{Y}]$ means that the matrix $\underline{X}-\underline{Y}$ is positive [semi] definite.

which again shows that $\underline{V}_L(t) \geq \underline{K}(t)$. Note that $\underline{V}_L(t)=\underline{K}(t)$ if and only if $\underline{L}(\tau)=\underline{L}^*(\tau)$ for all $\tau \varepsilon (t, t_1)$.

SUBOPTIMAL DESIGN TECHNIQUES - MOTIVATION

In order to implement the optimal control as a feedback law, it becomes necessary to compute $\underline{K}(t)$ using one of various numerical schemes. Conceptually, this task can be done in either an "on-line" or an "off-line" manner.

One may suggest to generate $\underline{K}(t)$, and in turn $\underline{L}^*(t)$, on line, while the system is in operation, as follows. Since $\underline{K}(t)$ is independent of the system state, the matrix $\underline{K}(t_0)$ may be precomputed (once $\underline{A}$, $\underline{B}$, $\underline{Q}$, and $\underline{R}$ have been specified) by integrating the Riccati equation backwards in time, starting from the boundary condition $\underline{K}(t_1)=\underline{F}$. One then stores only the elements of $\underline{K}(t_0)$ in the feedback controller, and, in real time, computes $\underline{K}(t)$ for $t>t_0$ by integrating Eq. 5 forward in time. Although this scheme is asthetically pleasing it is not feasible from a practical point of view since the solution $\underline{K}(t)$ is unstable in the forward-time direction.[2,3] In other words, a small numerical error in $\underline{K}(t_0)$ will manifest itself in a large error in $\underline{K}(t)$ for $t>t_0$, making it virtually impossible to accurately compute $\underline{K}(t)$ for $t>t_0$ from the knowledge of $\underline{K}(t_0)$.

An alternate approach to the implementation of $\underline{L}^*(t)$ is to precompute $\underline{K}(t)$ for $t_0<t<t_1$, by solving the Riccati equation in its stable direction (backwards in time), starting from $\underline{K}(t_1)=\underline{F}$. Once accomplished, the gain matrix $\underline{L}^*(t)=\underline{R}^{-1}(t)\underline{B}'(t)\underline{K}(t)$ is calculated and stored on tape in the feedback controller. The tape is played back in real time once the system begins operation. However, there are some engineering difficulties inherent in this approach. For multi-input, multi-output systems one is required to store a large number (depending on the dimension of $\underline{L}^*(t)$) of time-varying signals on tape. At time t_0 these signals must not only be "played-back" simultaneously but must be also played back in time-synchronization with one another and in synchronization with real time. The circuitry demanded by high-order systems can therefore become quite unwieldy, forcing one to consider more practical schemes for control implementation.

Note that similar considerations enter if one wishes to store $\underline{L}^*(t)$ in the memory of a digital computer. To do this one divides the interval $[t_0, t_1]$ into a large number of subintervals, say N, and stores the approximately constant values of $\underline{L}^*$. This requires a total of r.n.N memory locations, a number which may exceed the memory capacity of the available computer.

In the following sections we consider these problems and suggest a means of designing the linear feedback loop in a suboptimal fashion.

THE STRUCTURE OF THE SUBOPTIMAL CONTROL

In proposing a structure for a suboptimal regulator control, two things should be considered. First that it be of a form which readily lends itself to actual implementation, and second that it be of sufficient generality so that the optimal control can be approximated to any degree of accuracy. The suboptimal design scheme to be investigated consists of approximating the gain matrix $\underline{L}^*(t)$ over the interval of interest (t_0, t_1). In this manner the suboptimal control is a linear feedback law, so that it is not necessary to introduce nonlinear function generators into the feedback system. This is only reasonable since the system (1) is linear and the optimal control itself is a linear feedback law.

The proposed scheme uses an arbitrary set of linearly independent, scalar time functions $\alpha_j(t), j=1,2,\ldots,M$, defined for $t\varepsilon[t_0,t_1]$. The control input to the system (1) is restricted to be of the form

$$\underline{u}(\underline{x},t) = -\left(\sum_{j=1}^{M} \alpha_j(t)\underline{L}_j\right)\underline{x}(t) \tag{17}$$

or, equivalently, the gain matrix $\underline{L}(t)$ is constrained to be

$$\underline{L}(t) = \sum_{j=1}^{M} \alpha_j(t)\underline{L}_j \tag{18}$$

where $\underline{L}_j$, for $j=1,2,\ldots,M$, are arbitrary constant rxn matrices.

The matrices $\underline{L}_j$ are free to be chosen so as to make the suboptimal control law (17) close to the optimal control law $\underline{u}^*(\underline{x}(t),t)=-\underline{L}^*(t)\underline{x}(t)$ in some reasonable way. Note that as M increases, it is possible to choose the set of matrices $\{\underline{L}_j\}$ so that $\underline{L}(t)$ becomes an increasingly better approximation to $\underline{L}^*(t)$, albeit with an increasing complexity in $\underline{L}(t)$. This aspect is studied further in the sequel.

Compared to $\underline{L}^*(t)$, a feedback gain matrix of the form (18) is relatively easy to implement. Implementation requires only M synchronized tape-tracks on which the $\alpha_j(t)$ functions are stored (where M and $\alpha_j(t)$ are chosen a priori), and n.r.M constant gains which correspond to the values of the elements of the matrices $\underline{L}_j$, $j=1,\ldots,M$. In this manner, one can circumvent the engineering difficulties associated with storing a large number (n.r) of time-varying functions on tapes as required in the optimal solution.

THE SUBOPTIMAL LINEAR REGULATOR PROBLEM

In this section the problem of optimally selecting the matrices $\underline{L}_j$ is formulated in a precise manner.

Definition : Let M be a fixed integer and let $\alpha_j(t)$, $j=1,2,\ldots M$ be a given set of M linearly independent, scalar time-functions which are defined for $t\varepsilon[t_0,t_1]$. The matrix-valued function $\underline{L}(.)$ is said to be of class Λ_M if

$$\underline{L}(t) = \sum_{j=1}^{M} \alpha_j(t)\underline{L}_j \quad \text{for } t\varepsilon[t_0,t_1] \tag{19}$$

where $\underline{L}_j$, $j=1,2,\ldots,M$, are arbitrary rxn constant matrices. More succintly, Λ_M is the set

$$\Lambda_M \triangleq \{\underline{L}(.):\underline{L}(t) = \sum_{j=1}^{M} \alpha_j(t)\underline{L}_j \quad \text{for } t\varepsilon[t_0,t_1]\} \tag{20}$$

Note that Λ_M is a linear function space for if $\underline{L}_1(.)$ and $\underline{L}_2(.)\varepsilon\Lambda_M$, we have that $a\underline{L}_1(.) + b\underline{L}_2(.)\varepsilon\Lambda_M$ for all scalars a,b. Λ_M becomes a normed linear function space by defining the norm

$$||\underline{L}(.)|| \triangleq \left[\int_{t_0}^{t_1} ||\underline{L}(t)||^2 dt\right]^{1/2} \tag{21}$$

where $||\underline{L}(t)||$ is the matrix norm induced by the euclidean vector norm.

The manner in which the matrices $\underline{L}_j$ are to be determined will now be discussed. Let $\underline{L}(.)\varepsilon\Lambda_M$ and let $\underline{V}_L(t)$ be the cost-matrix (14) associated with $\underline{L}(t)$. Thus, if $\underline{x}(t_o)=\underline{x}_o$, the cost (3) resulting from applying $\underline{u}(\underline{x},t)=-\underline{L}(t)\underline{x}(t)$ to the system (1) is

$$J(\underline{x}_0,t_0;\underline{u}(.))\Big|_{\underline{u}=-\underline{L}(t)\underline{x}(t)} = \underline{x}_0'\underline{V}_L(t_0)\underline{x}_0 \tag{22}$$

It seems reasonable, inasmuch as the original control objective was to minimize $J(\underline{x}_0,t_0;\underline{u}(.))$, that the matrices $\underline{L}_j$ (or equivalently the gain matrix $\underline{L}(t)$) should be chosen so as to minimize the cost (22). However, this optimal choice of $\underline{L}_j$ will generally depend upon the initial state $\underline{x}_0$. This is undesirable, since in most regulator applications the initial state $\underline{x}_0$ is not known a priori.

The proposed alternative here is to *choose $\underline{L}(.)\varepsilon\Lambda_M$ to minimize a functional which is independent of the initial state $\underline{x}_0$. The functional chosen is the trace of $\underline{V}_L(t_0)$*, i.e.,

$$\mu(\underline{L}) = \mathrm{tr}[\underline{V}_L(t_0)] \tag{23}$$

The functional has the following physical interpretation. Suppose that the initial state $\underline{x}_0$ is a random variable which is uniformly distributed over the surface of an n-dimensional unit sphere; it is then possible to show that minimizing the cost (23) is equivalent to minimizing the expected value (over $\underline{x}_0$) of the cost (22). A further interpretation of $\mu(\underline{L})$ is that $\mu(\underline{L})/n$ is the *average* value of the cost (22) as $\underline{x}_0$ varies over the surface of the unit sphere.

A lower bound for $\mu(\underline{L})$ is readily obtained by taking the trace of both sides of Eq. 10 of Lemma 1, with the result that

$$\mathrm{tr}[\underline{K}(t_0)] \leq \mathrm{tr}[\underline{V}_L(t_0)] = \mu(\underline{L}) \quad \text{for all } \underline{L}(.)\varepsilon\Lambda_M \tag{24}$$

where, from Eq. 16, equality can hold if and only if $\underline{L}(t)=\underline{L}^*(t)$ for all $t\varepsilon[t_0,t_1]$.

In summary, the suboptimization problem to be considered is :

Given the integer M and the set of time-functions $\alpha_j(t)$, j=1,J,...,M. Determine the element $\underline{L}(.)\varepsilon\Lambda_M$ which minimizes $\mu(\underline{L})$.

Denote by $\underline{L}^o(.)$ the best suboptimal gain matrix, i.e.,

$$\underline{L}^o(.) = \arg\min_{\underline{L}(.)\varepsilon\Lambda_M} \{\mathrm{tr}\underline{V}_L(t_0)\} \tag{25}$$

and by

$$\mu^o = \mu(\underline{L}^o) = \min_{\underline{L}(.)\varepsilon\Lambda_M} \{\mathrm{tr}\underline{V}_L(t_0)\} \tag{26}$$

the best suboptimal cost.

In the next two sections the following two important aspects of the posed suboptimization problem will be examined:

1) Does $\underline{L}^o(t)\to\underline{L}^*(t)$ as $M\to\infty$
2) How does one determine $\underline{L}^o(.)$ given a fixed M?

PROPERTIES OF THE SUBOPTIMAL SOLUTION AS M→∞

For a prespecified set of scalar time-functions $\{\alpha_j(t)\}$ let $\underline{L}_M^o(.)\varepsilon\Lambda_M$ minimize $\mu(\underline{L})$. One then suspects that $\underline{L}_M^o(t)$ will well-approximate $\underline{L}^*(t)$ as M increases. In addition, if μ_M^o is the minimum value of $\mu(\underline{L})$ for $\underline{L}(.)\varepsilon\Lambda_M$, then one also expects $\mu_M^o\to\mathrm{tr}\underline{K}(t_0)$ as $M\to\infty$. These notions will now be investigated in a more precise mathematical framework.

Let $\alpha_1(t)$, $\alpha_2(t)$,... be an infinite sequence of real-valued scalar time-functions which are defined and square integrable on $[t_0,t_1]$. This class of functions is denoted by $\mathcal{L}^2[t_0,t_1]$.

Definition : *The sequence* $\{\alpha_j(t)\}$ *is complete* (in $\mathcal{L}^2[t_0,t_1]$) *if given any* $\beta(t)\varepsilon\mathcal{L}^2[t_0,t_1]$ *and any* $\varepsilon>0$ *there is a linear combination* $\beta_k(t)$ *of the form*

$$\beta_k(t) = \sum_{j=1}^{k} c_j\alpha_j(t) \tag{27}$$

where $k=k(\varepsilon)$ *and* $c_1,c_2,\ldots,c_k$ *are real numbers such that*

$$|\beta_k(.)-\beta(.)| \triangleq [\int_{t_0}^{t_1} |\beta_k(t)-\beta(t)|^2 dt]^{1/2} \leq \varepsilon \tag{28}$$

Assume that the sequence $\{\alpha_j(t)\}$ is complete in $\mathcal{L}^2[t_0,t_1]$.* For a given $M<\infty$, let Λ_M be the set (20) with $\alpha_1(t)$, $\alpha_2(t),\ldots,\alpha_M(t)$ being the first M elements of the sequence $\{\alpha_j(t)\}$. Let

$$\underline{L}_M^o(.) \triangleq \arg \min_{\underline{L}(.)\varepsilon\Lambda_M} \mu(\underline{L}) \tag{29}$$

and

$$\mu_M^o = \min_{\Lambda_M} \mu(\underline{L}) = \mu(\underline{L}_M^o) \tag{30}$$

denote the suboptimal gain matrix and the suboptimal cost, respectively, indicating the dependence of these quantities upon M. Clearly μ_M^o cannot increase with M, i.e.,

$$\mu_1^o \geqslant \mu_2^o \geqslant \ldots \tag{31}$$

since any element $\underline{L}(.)\varepsilon\Lambda_{M_1}$ is automatically an element of Λ_{M_2} for $M_2 > M_1$. Correspondingly

$$\Lambda_1 \subset \Lambda_2 \subset \ldots \tag{32}$$

It is proven in Ref. [6] that the following is true

Theorem 1 :

(i) $\lim_{M\to\infty} \mu_M^o = tr[\underline{K}(t_0)] = \mu(\underline{L}^*)$

(ii) $\lim_{M\to\infty} ||\underline{L}_M^o(.)-\underline{L}^*(.)|| = \lim_{M\to\infty} [\int_{t_0}^{t_1} ||\underline{L}_M^o(t)-\underline{L}^*(t)||^2 dt]^{1/2} = 0$

Hence, the intuitively expected asymptotic results discussed above are true. For a spacific set of *complete* functions $\{\alpha_j(t)\}$, μ_M^o converges to $tr[\underline{K}(t_0)]$ as $M\to\infty$; recall from Eq. 16 that $tr[\underline{K}(t_0)]$ is the greatest lower bound for $\mu(L)$. In addition, the sequence $\underline{L}_1^o(.)$, $\underline{L}_2^o(.),\ldots$, converges to a limit (in a mean square sense) and the limit is $\underline{L}^*(.)$. Therefore, by allowing greater complexity in the form of the gain matrix $\underline{L}(.)$, one can achieve a suboptimal system performance approaching optimality in return for the increased hardware requirements needed to implement a gain matrix of the form (19) as M increases.

Theorem 1 gives the results which are notably similar to those which appear in the study of the Ritz method in the calculus of variations. (see Ref. 7, Chapter 8). Borrowing the terminology indigenous to this method, it has been shown here that the sequence $\underline{L}_1^o(.)$, $\underline{L}_2^o(.),\ldots$ is a minimizing sequence for the functional $\mu(\underline{L})$ since $\mu_M^o\to\mu(\underline{L}^*)$. In addition, it has been proven here that the minimizing sequence has a *limit*. Generally, this is an extremely difficult task in most applications of the Ritz method, and

---

*Examples of such sequences are (1) $\alpha_j(t)= t^{(j-1)}$, $j=1,2,\ldots$
(2) $\alpha_j(t) = e^{(j-1)t}$ $j=1,2,\ldots$

depends on the detailed structural form of the functional to be minimized (in this case $\mu(\underline{L})$).

The speed of convergence of μ_M^o to $\mu(L^*)$ and of $\underline{L}_M^o(.)$ to $\underline{L}^*(.)$ depends both upon the origintal optimization problem itself and on the choice of the functions $\alpha_j(t)$. In any particular situation one would like to choose the set $\{\alpha_j\}$ so that linear combinations of the form (19) involving only a very small number of functions α_j will result in quite satisfactory approximations to $\underline{L}^*(.)$ and $\mu(\underline{L}^*)$. This is a difficult problem and remains a subject for further research.

NECESSARY CONDITIONS FOR THE SUBOPTIMALITY OF $\underline{L}^o(.)$

In this section, necessary conditions for $\underline{L}^o(.)\varepsilon\Lambda_M$ to minimize $\mu(\underline{L}) = \mathrm{tr}[\underline{V}_L(t_0)]$ are derived. It is assumed that M is fixed and that the functions $\alpha_j(t)$, j=1,2,...,M are given. Therefore, the task of determining $\underline{L}^o(.)$ is equivalent to determining the M constant matrices $\underline{L}_j$ such that $\mu(\underline{L})$ is minimized. The set of matrices $\underline{L}_j$ at which $\mu(\underline{L})$ attains its minimum is denoted by $\{\underline{L}_j^o,\ j=1,2,\ldots,M\}$, i.e.

$$\underline{L}^o(t) = \sum_{j=1}^{M} \alpha_j(t)\underline{L}_j^o \quad \text{for } t\varepsilon[t_0,t_1] \tag{33}$$

and the cost matrix associated with $\underline{L}^o(t)$ is

$$\underline{V}_0(t) = \underline{V}_L(t)\Big|_{\underline{L}(t)=\underline{L}^o(t)} \tag{34}$$

For a fixed value of j, the functional $\mathrm{tr}[\underline{V}_L(t_0)]$ is continuously differentiable with respect to the elements of the matrix $\underline{L}_j$. Therefore, $\underline{L}_j^o$ must satisfy the necessary condition

$$\frac{\partial \mathrm{tr}\underline{V}_L(t_0)}{\partial(\underline{L}_j^o)^{k\ell}} = 0 \quad \text{for } j=1,2,\ldots,M \quad k=1,\ldots,r;\ \ell=1,\ldots,n \tag{35}$$

where $(L_j)^{k\ell}$ is the $k\ell$-th element of the matrix $\underline{L}_j$.

Introducing the concept of a *gradient matrix* as defined in Refs. 8 and 9, Eq. 35 may be written in a more compact form as

$$\frac{\partial \mathrm{tr}\underline{V}_L(t_0)}{\partial \underline{L}_j^o} = \underline{0} \quad \text{for } j=1,2,\ldots,M \tag{36}$$

One then has

Theorem 2 : *(Necessary conditions for suboptimality)*
If $\underline{L}^o(.)\varepsilon\Lambda_M$ *minimizes* $\mathrm{tr}[\underline{V}_L(t_0)]$, *then for all* j=1,2,...,M

$$\underline{0} = \int_{t_0}^{t_1} \alpha_j(t)[\underline{R}(t)\underline{L}^o(t)-\underline{B}'(t)\underline{V}_0(t)]\underline{\Phi}_0(t,t_0)\underline{\Phi}_0'(t,t_0)dt = -\frac{1}{2}\,\frac{\partial \mathrm{tr}\underline{V}_L(t_0)}{\partial \underline{L}_j^o} \tag{37}$$

where $\underline{\Phi}_0(t,t_0)$ *denotes the transition matrix corresponding to* $[\underline{A}(t)-\underline{B}(t)\underline{L}^o(t)]$, *i.e.*,

$$\frac{d}{dt}\underline{\Phi}_0(t,t_0) = [\underline{A}(t)-\underline{B}(t)\underline{L}^o(t)]\underline{\Phi}_0(t,t_0)\ ;\ \underline{\Phi}_0(t_0,t_0) = \underline{I} \tag{38}$$

Proof : The method of Ref. 6, Appendix F, for computing the gradient matrix of a trace function will be used to obtain $\dfrac{\partial \mathrm{tr}\underline{V}_L(t_0)}{\partial \underline{L}_j^o}$. Define, for fixed j,

$$\underline{L}^\varepsilon(t) = \underline{L}^o(t)+\varepsilon\alpha_j(t)\Delta\underline{L}_j \quad \text{for } t_0 \leq t \leq t_1 \tag{39}$$

where $\epsilon\Delta\underline{L}_j$ denotes a small deviation from $\underline{L}_j^o$. If $\underline{V}_\epsilon(t)$ is the cost matrix associated with $\underline{L}^\epsilon(t)$, by the linearity of the tract functional,

$$\mathrm{tr}\underline{V}_\epsilon(t_0)-\mathrm{tr}\underline{V}_0(t_0) = \mathrm{tr}[\underline{V}_\epsilon(t_0)-\underline{V}_L(t_0)] \tag{40}$$

Application of Eq. A.9 of Appendix A yields

$$\underline{V}_\epsilon(t_0)-\underline{V}_0(t_0)= \int_{t_0}^{t_1} \underline{\Phi}'_\epsilon(t,t_0)[(\underline{L}^\epsilon-\underline{L}^o)'\underline{R}(\underline{L}^\epsilon-\underline{L}^o)-(\underline{L}^\epsilon-\underline{L}^o)'(\underline{B}'\underline{V}_0-\underline{RL}^o) \\ -(\underline{B}'\underline{V}_0-\underline{RL}^o)'(\underline{L}^\epsilon-\underline{L}^o)]\underline{\Phi}_\epsilon(t,t_0)dt \tag{41}$$

where $\underline{\Phi}_\epsilon(t,t_0)$ is the transition matrix corresponding to $[\underline{A}(t)-\underline{B}(t)\underline{L}^\epsilon(t)]$. Substituting Eq. 40 into Eq. 39 and using the definition of $\underline{L}^\epsilon(t)$ yields

$$\mathrm{tr}[\underline{V}\ (t_0)-\underline{V}_0(t_0)] =$$

$$\mathrm{tr}\int_{t_0}^{t_1} \underline{\Phi}'_\epsilon(t,t_0)[\epsilon^2\alpha_j^2(t)(\Delta\underline{L}_j)'\underline{R}(\Delta\underline{L}_j)-2\epsilon\alpha_j(t)(\underline{B}'\underline{V}_0-\underline{RL}^o)'(\Delta\underline{L}_j)]\underline{\Phi}_\epsilon(t,t_0)dt \tag{42}$$

But to first order in ϵ, for $t\epsilon[t_0,t_1]$,

$$\underline{\Phi}_\epsilon(t,t_0)=\underline{\Phi}_0(t,t_0)+\epsilon\underline{\Phi}_0(t,t_0)\int_{t_0}^{t}\alpha_j(\tau)\underline{\Phi}_0(t_0,\tau)\underline{B}(\tau)(\Delta\underline{L}_j)\underline{\Phi}_0(\tau,t_0)d\tau \tag{43}$$

Hence, to first order in ϵ, $\mathrm{tr}[\underline{V}_\epsilon(t_0)-\underline{V}_0(t_0)]$ becomes, using the cyclic property of the trace functional,

$$\mathrm{tr}[\underline{V}_\epsilon(t_0)-\underline{V}_0(t_0)] = -2\epsilon\mathrm{tr}[\int_{t_0}^{t_1}\alpha_j(t)\underline{\Phi}_0(t,t_0)\underline{\Phi}'_0(t,t_0)[\underline{B}'(t)\underline{V}_0(t)-\underline{R}(t)\underline{L}^o(t)]' \\ (\Delta\underline{L}_j)dt] \tag{44}$$

This implies (Ref. 6, Appendix F) that

$$\left.\frac{\partial\mathrm{tr}\underline{V}_L(t_0)}{\partial\underline{L}_j}\right|_{\underline{L}_j=\underline{L}_j^o}=-2\int_{t_0}^{t_1}\alpha_j(t)[\underline{B}'(t)\underline{V}_0(t)-\underline{R}(t)\underline{L}^o(t)]\underline{\Phi}_0(t,t_0)\underline{\Phi}'_0(t,t_0)dt \tag{45}$$

Since this matrix must equal to the zero matrix if $\underline{L}^o(.)$ is to minimize $\mathrm{tr}[\underline{V}_L(t_0)]$ the desired result follows. Q.E.D.

Theorem 2 provides only necessary conditions for $\underline{L}^o(.)\epsilon\Lambda_M$ to locally minimize $\mathrm{tr}[\underline{V}_L(t_0)]$. Sufficiency of this condition has not been shown, nor has it been shown that for a given set of functions $\{\alpha_j(t)\}$ there exists a unique $\underline{L}^o(.)\epsilon\Lambda_M$ satisfying Eq. 37. These matters are currently being investigated as are iterative schemes for the determination of $\underline{L}^o(.)$.

A SECOND-ORDER EXAMPLE

In this section a simple second-order example is presented which numerically illustrates the suboptimization scheme. The system considered is time-invariant and is characterized by the equation

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0\\ 1\end{bmatrix}u \tag{46}$$

The initial and final times are taken as $t_0=0$ and $t_1=2$, respectively, and the matrices $\underline{F}$, $\underline{Q}$, $\underline{R}$, in the cost functional are chosen to be

$$\underline{F} = \underline{0},\ \underline{Q} = \begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix},\ \underline{R} = r = 1 \tag{47}$$

The time-varying optimal gain matrix $\underline{L}^*(t)=\underline{b}'\underline{K}(t)$ is the row vector

$$\underline{L}^*(t) = [k_{12}(t)k_{22}(t)] \quad \text{for } t\epsilon[0,2] \tag{48}$$

and the functions $k_{12}(t)$ and $k_{22}(t)$ are shown in Fig. 1.

The cost $\mu(\underline{L}^*)$ is computed to be

$$\mu(\underline{L}^*) = \mathrm{tr}[\underline{K}(0)] = 2.5144 \tag{49}$$

Suboptimal control is considered for the case M=1 and $\alpha_1(t)=1.0$, so that the gain matrix is constrained to be constant over $[0,2]$. The suboptimization problem is to determine the matrix $\underline{L}_1^o$ which minimizes $\mu(\underline{L}) = \mathrm{tr}[V_{\underline{L}}(0)]$.

Equation 37 was solved numerically by using a gradient scheme (see Ref. 6), with the result that $\underline{L}_1^o$ turned out to be

$$\underline{L}_1^o = [.8095 \quad 1.1668] \tag{50}$$

The suboptimal cost $\mu(\underline{L}^o)$ was found to be

$$\mu(\underline{L}^o) = 2.6283 \tag{51}$$

which within five percent of $\mu(\underline{L}^*)$ (see Eq. 49). The elements of $\underline{L}_1^o$ are also shown in Fig. 1.

REMARKS

The basic concept underlying the suboptimal design scheme is that one first specifies a simple time-varying structure for the feedback gains. By virtue of the choice of $\mu(L)$, suboptimal control then has the important property of being independent of the initial state x_0. This allows the suboptimal gains to be computed prior to system run time so that no computations need be done on-line.*

Regarding the theoretical investigations, the results of Theorem 1 showed certain similarities between the suboptimization problem and the Ritz method. At present the Ritz method is familiar to physicists and is a standard direct method in the calculus of variations. However, it does not seem to have been applied to the solution of optimal (or suboptimal) control problems; Theorem 1 suggests that it may be fruitful to do so.

Included in the list of yet unsolved problems are methods for choosing the set of functions $\{\alpha_j(t)\}$. Criteria for their choice are largely subjective, especially if one chooses to generate them on-line. In a later paper the specific, yet practical, case in which the gain matrix $\underline{L}(t)$ is constrained to be piecewise-constant over $[t_0,t_1]$ will be investigated. This corresponds to the functions $\alpha_j(t)$ being piecewise-constant with values either 0 or 1.

Finally, note that although the analysis was conducted in a control-theoretic context, the results of the study apply equally well to linear filter problems in which the filter's time-varying structure is constrained. In this case, one specifies a structure for the time-varying gains of the Kalman filter so as to speed up the required real time computations.

CONCLUSIONS

The feasibility of taking practical engineering constraints into consideration when designing linear regulator systems has been demonstrated. This was done by prespecifying the structural form of time-varying feedback gains, while leaving various free parameters to be chosen in an optimal fashion. It was shown that under certain completeness assumptions, the solution of the resulting *suboptimal* linear regulator problem approached the solution of the *optimal* linear regulator problem. Finally, necessary conditions for the determination of the suboptimal solution were derived. A simple example was presented which indicated that the approach yields results that can be used in an actual suboptimal design.

---

*If the initial state, or some of its components, are known, the necessary conditions (37) can be slightly modified to include the additional information (see Ref. 6, p. 87).

ACKNOWLEDGMENT

This paper is based on a dissertation submitted by the first author on January, 1967 to the MIT Electrical Engineering Department in partial fulfillment of the requirements for the degree Doctor of Philosophy. The research was conducted at the MIT Electronic Systems Laboratory with support extended by the NASA Electronics Research Center under research grant NGR-22-009(124), DSR project No. 76265.

REFERENCES

1. Athans, M., and Falb, P.L., Optimal Control: *An Introduction to the Theory and its Applications*, McGraw-Hill Book Co., New York, 1966.
2. Kalman, R.E., "Contributions to the Theory of Optimal Control," *Bol. Soc.Mat. Mexico*, 1960, pp. 102-119.
3. Kleinman, D.L., "On the Linear Regulator Problem and the Matrix Riccati Equation," M.I.T. Electronic Systems Laboratory Report ESL-R-271, Cambridge, Mass., June, 1966.
4. Kalman, R.E., and Bucy, R.S., "New Results in Linear Filtering and Prediction Theory," *J. Basic Engineering, Trans. ASME*, Vol. 83, 1961, pp. 95-108.
5. Falb, P.L. and Kleinman, D.L., "Remarks on the Infinite Dimensional Riccati Equation," IEEE Trans. on Auto. Control, Vol. 11, No.3, 1966, pp. 534-537.
6. Kleinman, D.L., "Suboptimal Design of Linear Regulator Systems Subject to Computer Storage Limitations," Ph.D. thesis, Dept. of Electrical Engineering, MIT, February 1967 also MIT Electronic Systems Laboratory Report ESL-R-297, Cambridge, Mass., February, 1967.
7. Gelfand, I.M. and Fomin, S.V., *Calculus of Variations*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1963.
8. Athans, M. and Schweppe, F.C., "Gradient Matrices and Matrix Calculations," M.I.T. Lincoln Lab. Technical Note 1965-53 (unpublished), Lexington, Mass., 1965.
9. Athans, M., "The Matrix Minimum Principle," submitted for publication to *Information and Control*, July, 1967.

APPENDIX A

COST MATRICES FOR LINEAR REGULATOR PROBLEMS

The cost matrix $\underline{V}_L(t)$ associated with a gain matrix $\underline{L}(t)$ is given by Eq. 14, and is the unique solution of Eq. 9 satisfying $\underline{V}_L(t_1)=\underline{F}.\underline{V}_L(t)$ has the property that

$$J(\underline{x},t;\underline{u}(.))\Big|_{\underline{u}=-\underline{L}(t)\underline{x}(t)} = \underline{x}'\underline{V}_L(t)\underline{x} \qquad (A.1)$$

Suppose that one has two different control laws $\underline{u}_1=-\underline{L}_1(t)\underline{x}(t)$ and $\underline{u}_2 = -\underline{L}_2(t)\underline{x}(t)$. For a given $\underline{x}\varepsilon R_n$, the cost difference between applying $\underline{u}_1$ as opposed to $\underline{u}_2$ is

$$J(\underline{x},t;\ \underline{u}_1(.))-J(\underline{x},t;\underline{u}_2(.))=\underline{x}'[\underline{V}_1(t)-\underline{V}_2(t)]\underline{x} \qquad (A.2)$$

where $\underline{V}_1(t)$ and $\underline{V}_2(t)$ are the cost matrices associated with $\underline{L}_1(t)$ and $\underline{L}_2(t)$ respectively. Therefore, studying the cost difference between control laws is tantamount to studying the difference in their associated cost matrices.

Several expressions are derived below for $\underline{V}_1(t)$ and $\underline{V}_2(t)$. The notation $\underline{A}_i(t)=\underline{A}(t)-\underline{B}(t)\underline{L}_i(t)$ for i=1,2 is used. and $\underline{\Phi}_i(\tau,t)$ denotes the state transition matrix corresponding to $\underline{A}_i(t)$.

The matrix $\underline{V}_1(t)$ satisfies the differential equation

$$\dot{\underline{V}}_1(t)=-\underline{A}_1'(t)\underline{V}_1(t)-\underline{V}_1(t)\underline{A}_1(t)-\underline{Q}(t)-\underline{L}_1'(t)\underline{R}(t)\underline{L}_1(t) \tag{A.3}$$

Writing $\underline{A}_1(t)=\underline{A}_2(t)-\underline{B}(t)[\underline{L}_1(t)-\underline{L}_2(t)]$ yields

$$\begin{aligned}\dot{\underline{V}}_1(t)=&-\underline{A}_2'(t)\underline{V}_1(t)-\underline{V}_1(t)\underline{A}_2(t)-\underline{Q}(t)-\underline{L}_1'(t)\underline{R}(t)\underline{L}_1(t)\\&+[\underline{L}_1(t)-\underline{L}_2(t)]'\underline{B}'(t)\underline{V}_1(t)+\underline{V}_1(t)\underline{B}(t)[\underline{L}_1(t)-\underline{L}_2(t)]\end{aligned} \tag{A.4}$$

But on the other hand, $\underline{V}_2(t)$ satisfies

$$\dot{\underline{V}}_2(t) = -\underline{A}_2'(t)\underline{V}_2(t)-\underline{V}_2(t)\underline{A}_2(t)-\underline{Q}(t)-\underline{L}_2'(t)\underline{R}(t)\underline{L}_2(t) \tag{A.5}$$

Subtracting Eq. A.5 from A.4 yields, writing $\underline{V}(t)=\underline{V}_1(t)-\underline{V}_2(t)$.

$$\begin{aligned}\delta\dot{\underline{V}}(t)=&-\underline{A}_2'(t)\delta\underline{V}(t)-\delta\underline{V}(t)\underline{A}_2(t)+\underline{L}_2'(t)\underline{R}(t)\underline{L}_2(t)-\underline{L}_1'(t)\underline{R}(t)\underline{L}_1(t)\\&+[\underline{L}_1(t)-\underline{L}_2(t)]'\underline{B}'(t)\underline{V}_1(t)+\underline{V}_1(t)\underline{B}(t)[\underline{L}_1(t)-L_2(t)]\end{aligned} \tag{A.6}$$

Since $\underline{V}_1(t_1)=\underline{V}_2(t_1)=\underline{F}$, $\delta\underline{V}(t_1)=\underline{0}$. Now add and subtract the term $(\underline{L}_1-\underline{L}_2)'\underline{RL}_2+\underline{L}_2'R(\underline{L}_1-\underline{L}_2)$ from the right hand side of Eq. A.6 to obtain

$$\begin{aligned}\delta\dot{\underline{V}}(t)=&-\underline{A}_2'(t)\delta\underline{V}(t)-\delta\underline{V}(t)\underline{A}_2(t)-[\underline{L}_1(t)-L_2(t)]'\underline{R}(t)[\underline{L}_1(t)-\underline{L}_2(t)]\\&+[\underline{L}_1(t)-\underline{L}_2(t)]'[\underline{B}'(t)\underline{V}_1(t)-\underline{R}(t)\underline{L}_2(t)]+[\underline{B}'(t)\underline{V}_1(t)-R(t)\underline{L}_2(t)]'\\&[\underline{L}_1(t)-\underline{L}_2(t)]\end{aligned} \tag{A.7}$$

The solution of Eq. A.7 with the boundary condition $\delta\underline{V}(t_1)=0$ is given by

$$\begin{aligned}\delta\underline{V}(t)=\int_t^{t_1}&\underline{\Phi}_2'(\tau,t)[\underline{L}_1-\underline{L}_2)'\underline{R}(\underline{L}_1-\underline{L}_2)-(\underline{L}_1-\underline{L}_2)'(\underline{B}'\underline{V}_1-\underline{RL}_2)\\&-(\underline{B}'\underline{V}_1-\underline{RL}_2)'(\underline{L}_1-\underline{L}_2)]\underline{\Phi}_2(\tau,t)d\tau\end{aligned} \tag{A.8}$$

where, for ease of notation in the integrand, $\underline{L}_1\equiv\underline{L}_1(t)$, etc.

By interchanging subscripts in Eq. A.8 one obtains an expression for $\underline{V}_2(t)-\underline{V}_1(t)$. Multiplying this newly found expression by -1 yields, after some algebraic manipulation, another formula for $\delta\underline{V}(t)=\underline{V}_1(t)-\underline{V}_2(t)$, namely

$$\begin{aligned}\delta\underline{V}(t)=\int_t^{t_1}&\underline{\Phi}_1'(\tau,t)[(\underline{L}_1-\underline{L}_2)'\underline{R}(\underline{L}_1-\underline{L}_2)-(\underline{L}_1-\underline{L}_2)'(\underline{B}'\underline{V}_2-\underline{RL}_2)\\&-(\underline{B}'\underline{V}_2-\underline{RL}_2)'(\underline{L}_1-\underline{L}_2)]\underline{\Phi}_1(\tau,t)d\tau\end{aligned} \tag{A.9}$$

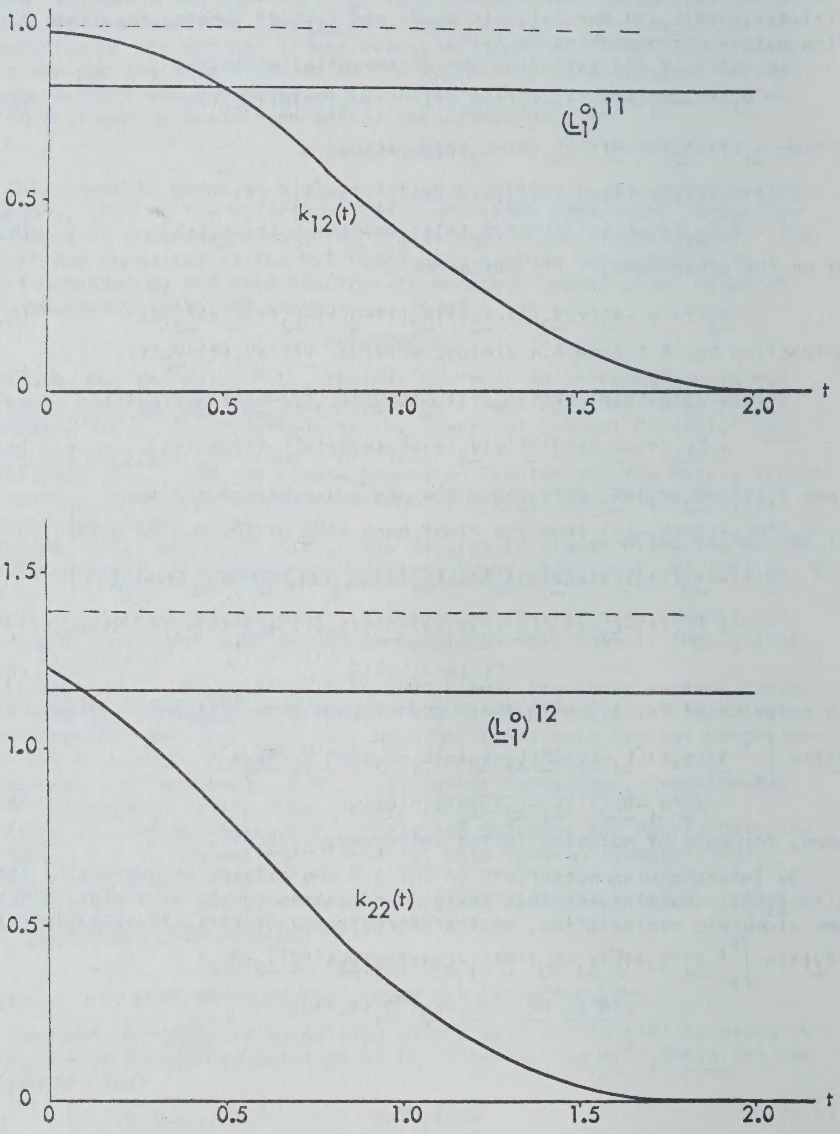


Fig. 1 Optimal and Suboptimal Gains for Second-Order Example
$M=1$, $a_1(t)=1$, $t_0=0$, $t_1=2$
--- Optimal Gains for $t_1 \to \infty$

GRAPHS WITH DEGENERACY IN BASIC CIRCUIT MATRICES

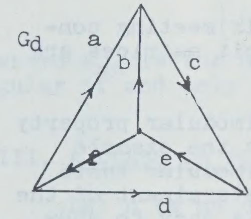
P. M. Lin
School of Electrical Engineering
Purdue University
Lafayette, Indiana

ABSTRACT

This paper presents two classes of graphs with the property that a set of circuits which are independent when the graphs are directed, become dependent when the graphs are undirected. The implication of the results from circuit theory point of view is discussed.

I. INTRODUCTION

Let B_a be the circuit matrix of a connected directed graph G_d of v vertices and e edges.[1] The rank of B_a, denoted by $r(B_a)$, is equal to $e-v+1=\mu$. Any $\mu \times e$ submatrix of B_a is called a basic circuit matrix if the submatrix has rank μ. (The rows of such a submatrix form a basis for the row space of B_a.) Let B_d be a basic circuit matrix of G_d. A matrix B_n is now derived from B_d by replacing all -1's by 1's, and the rank of B_n is determined using modulo 2 algebra. For practically all cases we have $r(B_d)=r(B_n)$. An exceptional case cited by Seshu and Reed [1] is repeated in Fig. 1 below.



$$B_d = \begin{array}{c} \begin{matrix} a & b & c & d & e & f \end{matrix} \\ \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & -1 & 0 & -1 \\ 1 & -1 & 0 & -1 & -1 & 0 \end{bmatrix} \end{array}$$

$$B_n = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \pmod 2$$

Fig. 1

It is a simple matter to show that B_d has rank 3 while B_n has rank 2. Therefore the graph of Fig. 1 has a basic circuit for which $r(B_d) \neq r(B_n)$. Degeneracy of this kind is uncommon, and therefore has not attracted much attention. It is the purpose of this paper to investigate whether the graph of Fig. 1 is the only pathological case or there are many of them. Obviously, any graph containing the graph of Fig. 1 as a subgraph (where every edge may be replaced by a series connection of edges) also has a basic circuit matrix with the mentioned degeneracy. We shall exclude such trivial extensions from our consideration. Our problem may now be stated precisely as follows:

---

[1] Unless stated otherwise, the terminoly and notations used here are the same as those in Reference [1].

Find a directed graph G_d which has a basic circuit matrix B_d satisfying the following two conditions

(a) $r(B_d) \neq r(B_n)$

(b) Any m rows of B_d, $m < \mu$, which are independent by the definition of a basic circuit matrix, will remain independent when all -1's are replaced by 1's and modulo 2 algebra is used in place of the ordinary algebra.

II. SOME PRELIMINARY CONSIDERATIONS

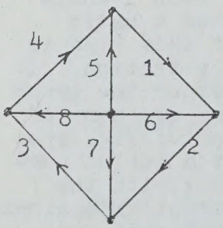
A matrix with all subdeterminants equal to 1, -1, or 0 is called a unimodular matrix.[1] Properties of unimodular matrices have been studied by a number of researchers [1] - [2] . It can be shown that B_d cannot meet condition (a) if it is unimodular.[2] Some known types of unimodular matrices which arise in circuit analysis may be found in [3] and [4] . We state below without proof those which have direct bearing on the present dtudy of basic circuit matrices:

The fundamental circuit matrix B_f corresponding to any tree is a unimodular matrix.

For a planar directed graph, the basic circuit matrix with rows corresponding to the "meshes" of the graph is a unimodular matrix.

Therefore in looking for a basic circuit matrix meeting condition (a), the above two types of basic circuit matrices are to be excluded from consideration.

Although condition (a) implies the nonunimodular property of B_d, the converse is not true. Fig. 2 shows the example used by Cederbaum [5] to illustrate a non-unimodular basic circuit matrix. B_d is not unimodular (the determinant of the first four columns is 3) and has rank 4. It is easy to show that B_n also has rank 4 (Property 1 below). Thus, non-unimodular property of a matrix does not imply the rank degeneracy given by (a).



$$B_d = \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \left[\begin{array}{cccccccc} 1 & 1 & 1 & 0 & 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 1 & -1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & -1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & -1 & 1 \end{array}\right] \end{array}$$

Fig. 2

---

[1] Unimodular matrices are designated as E-matrices in [2], and as Θ matrices in [4].

[2] See, for example, Chap. 5 of Reference [1].

Properties of some special matrices which have been found useful in the present study are stated below. The proofs are simple and hence omitted.

Property 1. Let T be a zero-axial matrix (square) of order n and with all off-diagonal elements equal to 1.

$$T = \begin{bmatrix} 0 & 1 & 1 & \cdots\cdots & 1 \\ 1 & 0 & 1 & \cdots\cdots & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & 1 & \cdots\cdots & 0 & 1 \\ 1 & 1 & \cdots\cdots & 1 & 0 \end{bmatrix} \qquad (1)$$

Then $|T| = (-1)^{n-1}(n-1)$. It follows that T is nonsingular for any $n \geq 2$.

Property 2. In mod 2 algebra, the zero-axial matrix T defined by (1) is nonsingular if and only if n is even.

Property 3. Let P be a square matrix of the form

$$P = \left[\begin{array}{c|cccc} a_{11} & & & & \\ a_{21} & & & & \\ \vdots & & U_{nn} & & \\ & & & & \\ a_{n1} & & & & \\ \hline a_{(n+1)1} & -1 & -1 & \cdots & -1 \end{array}\right] \qquad (2)$$

where U_{nn} is the unit matrix of order n. Then P is nonsingular if and only the sum of the first column is not zero.

III. A CLASS OF PLANAR GRAPHS

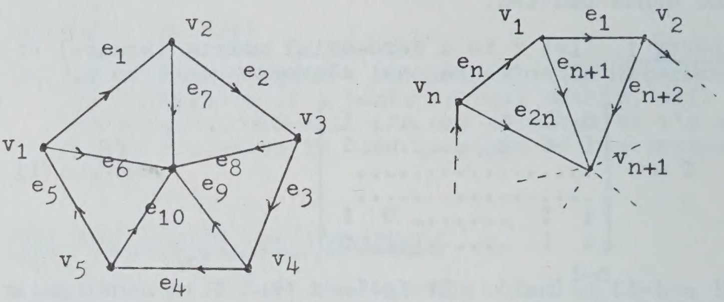
Let n be an odd integer greater than or equal to 3. We define a class of planar directed graphs S_1 with 2n edges and n+1 vertices by the following incidence relationships:[1]

$$e_i = \begin{cases} (v_i, v_{i+1}) & \text{for } i=1,2,\ldots\ n-1 \\ (v_n, v_1) & \text{for } i=n \\ (v_{i-n}, v_{n+1}) & \text{for } i=n+1,\ \ldots\ 2n \end{cases} \qquad (3)$$

Fig. 3(a) shows a graph of this class for n=5. Fig. 3(b) shows a part of the graph for the general case. The rank of the circuit matrix B_a is equal to $e-v+1=2n-(n+1)+1=n$.

---

[1] By $e_i = (v_j, v_k)$ we mean that v_j is the initial vertex and v_k is the terminal vertex of the directed edge e_i.



(a) (b)

Fig. 3

There exists an nxe submatrix of B_a of the following form

$$B_d = \left[\; B_1 \;\vdots\; B_2 \;\right]$$

$$= \begin{array}{c} \begin{array}{ccccc|ccccc} e_1 & e_2 & & & e_n & e_{n+1} & e_{n+2} & & & e_{2n} \end{array} \\ \left[\begin{array}{ccccc|ccccc} 0 & 1 & \cdots\cdots & & 1 & 1 & -1 & 0\cdots & & 0 \\ 1 & 0 & 1\cdots & & 1 & 0 & 1 & -1 & 0\cdots & 0 \\ & \cdots\cdots & & & & & \cdots\cdots & & & \\ 1 & \cdots & 1 & 0 & 1 & 0 & 0 & \cdots & 1 & -1 \\ 1 & \cdots & 1 & 1 & 0 & -1 & 0 & \cdots & 0 & 1 \end{array}\right] \end{array} \qquad (4)$$

where B_1 is the zero-axial matrix T of Property 1, and every row of B_2 has exactly two nonzero elements. (Note that the structure of B_2 is completely determined by the above constraints.) B_2 is recognized to have the form of the complete incidence matrix of a directed circuit with n edges.

Since $B_1=T$, then $|B_1| \neq 0$ according to Property 1. Then B_d has rank n, and is truly a basic circuit matrix of G_d. Since n is odd, every column of B_n has an even number of 1's. In modulo 2 algebra, the n rows of B_n are dependent. The rank of B_n is less than n. Thus the matrix B_d given by (4) satisfies condition (a).

If any number of rows of B_2 are deleted, then the remaining m rows ($m < n$) of B_2 are indepedent. (This is implied by Theorem 5-1 of [1].) Obviously, the corresponding m rows of B_d are also independent. Now with all -1's in B_2 replaced by 1's, we obtain B_{2n}. B_{2n} has the form of the complete incidence matrix of an undirected circuit with n edges. Therefore, any m rows ($m < n$) of B_{2n} are independent (Theorem 4-4 of [1]), and so are the corresponding m rows of B_n. Thus the basic circuit matrix (4) satisfies condition (b) also.

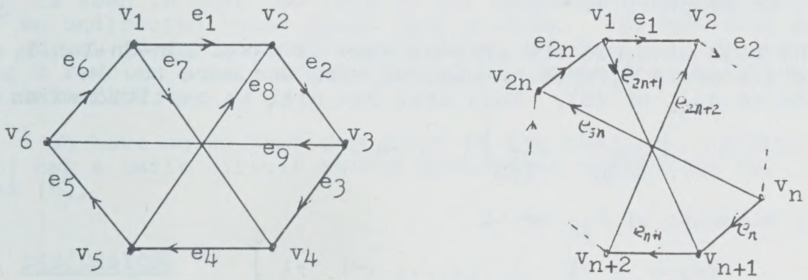
We have shown that any graph of the class S_1, defined by (3), has a basic circuit matrix satisfying conditions (a) and (b).

IV. A CLASS OF NONPLANAR GRAPHS

Let n again be an odd integer greater than or equal to 3. We define a class of nonplanar graphs S_2 with 3n edges and 2n vertices by the following incidence relationships:

$$e_i = \begin{cases} (v_i, v_{i+1}) & \text{for } i=1,2,\ldots\ 2n-1 \\ (v_{2n}, v_1) & \text{for } i=2n \\ (v_{i-2n}, v_{i-n}) & \text{for } i \text{ odd and } 2n < i \leq 3n \\ (v_{i-n}, v_{i-2n}) & \text{for } i \text{ even and } 2n < i \leq 3n \end{cases} \qquad (5)$$

Fig. 4(a) shows a graph of S_2 for n=3. Fig. 4(b) shows a part of the graph for the general case.



(a) (b)

Fig. 4

There exists an nxe submatrix of B_a of the form

$$B_{sub} = \left[B_{11} \;\vdots\; B_{12} \right] = \left[B_{11} \;\vdots\; U_{nn} \right]$$

$$= \begin{array}{c} \\ \\ \\ \\ \\ \end{array}\begin{array}{ccccccccc|ccc} e_1 & \cdots & e_n & e_{n+1} & \cdots & & & e_{2n} & & e_{2n+1} & \cdots & e_{3n} \\ \hline 0 & \cdots & 0 & 1 & & \cdots & & 1 & & 1 & 0 \cdots & 0 \\ 0 & 1 & \cdots & 1 & 1 & 0 & \cdots & 0 & & 0 & 1 & 0 \cdots\ 0 \\ & \cdots & & \cdots & \cdots & \cdots & & & & & \cdots & \\ 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & 1 & & 0 & \cdots & 1 \end{array} \qquad (6)$$

where B_{12} is an identity matrix of order n, and B_{11} has 1's and 0's only. (Note that the structure of B_{11} is completely determined by these constraints.) We next prove the following property of B_{11}.

Property 4. If the jth column of B_{11} has an even (odd) number of 1's, then the (j+1)th column has an odd (even) number of 1's.

Proof: Let p_j be the number of 1's of the jth column of B_{11}. Since every row of B_{11} has exactly n 1's, there are a total of n^2 1's in B_{11}. Adding up the number of 1's column by column, we have

$$p_1 + p_2 + \cdots\cdots + p_{2n} = n^2 \qquad (7)$$

From symmetry consideration, we have

$$\begin{aligned} p_1 &= p_3 = \cdots\cdots = p_{2n-1} \\ p_2 &= p_4 = \cdots\cdots = p_{2n} \end{aligned} \qquad (8)$$

From (7) and (8), it follows that

$$np_1 + np_2 = n^2$$

or $$p_1 + p_2 = n \qquad (9)$$

Since n is an odd integer, obviously p_2 must be even (odd) if p_1 is odd (even). This fact together with Equations (8) result in Property 4.

The rank of B_a in the present case is $e-v+1 = 3n-2n+1 = n+1$. To obtain a basic circuit matrix, we need one more row in addition to B_{sub} of (6). This last row will be partitioned as

$$\begin{bmatrix} \underset{1\times 2n}{B_{21}} & \underset{1\times n}{B_{22}} \end{bmatrix}$$

Let all elements of B_{22} be -1

$$B_{22} = \begin{bmatrix} -1 & -1 & \cdots\cdots -1 & -1 \end{bmatrix}$$

There are two possible forms of B_{21} which can be associated with the above B_{22}. They are

$$B_{21} = \begin{bmatrix} 1 & 0 & 1 & 0 & \cdots\cdots & 1 & 0 \end{bmatrix} \qquad (10)$$

or $$B_{21} = \begin{bmatrix} 0 & -1 & 0 & -1 & \cdots\cdots & 0 & -1 \end{bmatrix} \qquad (11)$$

Note that in either (10) or (11), zeros elements alternate with nonzero elements. A basic circuit matrix for the graphs of S_2 is now chosen to be

$$B_d = \left[B_1 \;\vdots\; B_2 \right] = \left[\begin{array}{c|c} B_{11} & B_{12} \\ B_{21} & B_{22} \end{array}\right] = \left[\begin{array}{c|c} B_{11} & U \\ B_{21} & B_{22} \end{array}\right] \qquad (12)$$

where either (10) or (11) is used as B_{21} so as to make every column of B_1 contain an even number of nonzero elements.

We shall now show that B_d as given by (12) satisfies conditions (a) and (b). B_2 together with any one column from B_1 assume the form of the matrix P of Property 3. The sum of any column of B_1 is

$$\tfrac{1}{2}(n+1) \qquad \text{for } n = 5, 9, 13, \ldots\ldots$$

or $$\tfrac{1}{2}(n-1) \qquad \text{for } n = 3, 7, 11, \ldots\ldots$$

which is always nonzero for $n \geq 3$. Therefore, according to Property 3, P has rank n+1. Then B_d has rank n+1 too. Now if all -1's are replaced by 1's, then every column of B_n has an even number of 1's. In modulo 2 algebra, the sum of all n+1 rows of B_n is a zero row. Therefore the rank of B_n is less than n+1. Thus $r(B_d) \neq r(B_n)$, condition (a) is met.

We observe that B_2 of (12) has the form of the complete incidence matrix of a directed "star tree" with n edges. If any number of rows of B_2 are deleted, then the remaining m rows ($m < n+1$) are independent , and so are the correspond-m rows of B_d (Theorem 5-1 of [1]). With all -1's replaced by 1's, we obtain

$$B_n = \begin{bmatrix} B_{1n} & B_{2n} \end{bmatrix}$$

B_{2n} is seen to have the form of the complete incidence matrix of an undirected "star tree" with n edges. For the same reason as above, if any number of rows of B_n are deleted, the remaining m rows ($m < n+1$) are independent. Therefore condition (b) is satisfied.

We have shown that any graph of the class S_2, defined by (5) has a basic circuit matrix satisfying conditions (a) and (b).

V. DISCUSSIONS

We have presented two classes of directed graphs with the property that each graph has a basic circuit matrix B_d such that $r(B_d) \neq r(B_n)$. From the symmetry of these graphs, and the way each circuit is selected, it is thought that group theory may be profitably used to generate much wider classes of graphs with the property $r(B_d) \neq r(B_n)$. The subject is under investigation.

Although our discussion has been restricted to basic circuit matrices, the results of Sec. III may be readily extended to basic cutset matrices Q_d by constructing the dual graphs. Such extension, however, is not applicable in the case of nonplanar graphs of Sec. Iv. We also point out that the directions of any edges or any circuits in the graphs of Sec. III and IV may be reversed without affecting the final results.

At present we know some sufficient conditions, and also some necessary conditions on B_d in order to have $r(B_d)=r(B_n)$. But the necessary and sufficient condition for $r(B_d)=r(B_n)$ has yet to be found.

In circuit analysis we use B_d (or Q_d) to formulate the equilibrium equations. Linear dependence or independence of the rows of B_n (with respect to mod 2 algebra) is of no consequence for such purposes. As we have just seen, in many cases the rank of B_n (mod 2) may not be equal to the rank of B_d. Theorems for (1,0) matrices using mod 2 algebra, which are useful in the study of undirected graphs, do not extend automatically to (1,-1,0) matrices which are associated with directed graphs. One has to prove the corresponding theorems again for directed graphs, although in many cases the proofs are similar to those for undirected graphs. It is felt that as far as the applications to circuit analysis is concerned, (1,-1,0) matrices should be given much more attention than (1,0) matrices and mod 2 algebra.

REFERENCES

1. S. Seshu and M. B. Reed, Linear Graphs and Electrical Networks, Reading Mass.: Addison-Wesley, 1961.

2. I. Cederbaum, "Matrices all of whose elements and subdeterminants are 1, -1, 0," J. Math. and Phys. Vol. 36, pp. 351-361, 1958.

3. I. Cederbaum, "Conditions for the impedance and admittance matrices of n-ports without ideal transformers," Proc. IEE (London), pt. C, Vol. 103, pp. 245-251, March 1958.

4. L. Weinberg, Network Analysis and Synthesis, New York: McGraw-Hill, 1962.

5. I. Cederbaum, "Invariance and mutual relations of electrical network determinants," J. Math. Phys., Vol. 34, pp. 236-244, January 1956.

ON MAXIMALLY DISTINCT TREES

G. KISHI and Y. KAJITANI
Department of Electronics
Tokyo Institute of Electronics
Tokyo Institute of Technology
Meguro-ku, Tokyo, Japan

ABSTRACT

A pair of trees of a connected linear graph G is said to be maximally distinct if the distance between them is maximum. As an immediate consequence, the following theorem is obvious:

Theorem 1: If a pair of trees has no common cords, then it is maximally distinct.

In order to extend Theorem 1, suppose that a pair of trees (t_1,t_2) has at least one common chord e_c. If there exists a minimum subgraph G_c of G satisfying (i) $G_c \ni e_c$, then G_c is said to be the K-subgraph of (t_1,t_2) with respect to common chord e_c. In terms of the definition, we can prove the rollowing theorem:

Theorem 2: Suppose that a pair of trees (t_A,t_B) has common chord e_{ci}, $(i = 1,2,\ldots,m)$. Then (t_A,t_B) is maximally distinct if and only if there exists one K-subgraph with respect to each common chord e_{ci}.

Under the assumption of Theorem 2, we can define the principal subgraph G_o as the union of all the K-subgraphs. Then we can obtain the following theorem:

Theorem 3: The principal subgraph is uniquely determined for all the pairs of maximally distinct trees.

A procedure yielding a pair of maximally distinct trees starting with an arbitrary pair of trees is presented.

As an application of the maximally distinct trees, we discuss the generation of all the trees of a connected graph G. Let (t_A,t_B) be a pair of maximally distinct trees of G. Now we choose the subset of edges defined by

$$E_o = \{e_1,e_2,\ldots,e_n\} = \{t_A - t_B\}.$$

We can define a subset of trees

$$T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ x_1,x_2,\ldots,x_n \end{pmatrix}$$

$$x_i = 0 \text{ or } 1 \quad (i = 1,2,\ldots,n)$$

as the collection of trees which contain e_i if $x_i=1$ but do not if $x_i=0$. It is clear that all the subsets form a classification of all the trees of G. Then the following theorem holds:

Theorem 4: If E_o satisfies

$$E_o = \{e_1, e_2, \ldots, e_n\} = \{t_A - t_B\}$$

where (t_A, t_B) is maximally distinct, then any subset $T\begin{pmatrix} e_1, e_2, \ldots, e_n \\ x_1, x_2, \ldots, x_n \end{pmatrix}$ is non-empty, and there exists at least one subset $T\begin{pmatrix} e_1, e_2, \ldots, e_n \\ y_1, y_2, \ldots, y_n \end{pmatrix}$ which contains only one tree.

This theorem provides a method of generating all the trees of G along a Hamilton circuit in the tree graph of G.

INTRODUCTION

In a connected linear graph, the distance between two vertices V_1 and V_2 can be defined as the minimum number of edges contained in any path connecting the vertices V_1 and V_2. If the pair of vertices (V_A, V_B) gives the maximum distance, the vertices are said to be maximally distinct.

Applying these definitions to the tree graph of a linear graph, we can easily define a pair of maximally distinct treet of the linear graph as a pair of trees whose distance is maximum in the tree graph. The purpose of this paper is to give some theorems concerning the fundamental properties of the maximally distinct trees with a direct application.

PRELIMINARIES

If an arbitrary pair of trees (t_1, t_2) is specified in a connected linear graph G, a common chord is defined as an edge which is in neither t_1 nor t_2. Let

b = the number of the edges of G,
v = the number of the vertices of G,
m = the number of the common chords with respect to (t_1, t_2),
$D(t_1, t_2)$ = the distance between t_1 and t_2 in the tree graph of G.

Then we can easily obtain

$$D(t_1, t_2) + m = b - v + 1. \qquad (1)$$

As an immediate consequence of (1), we can obtain the following theorem:

Theorem 1: If a pair of trees (t_A, t_B) has no common chords, then (t_A, t_B) is maximally distinct.

Thus, for example, the pair of trees (t_1, t_2) shown in Fig. 1 is maximally distinct.

MAXIMALLY DISTINCT TREES

In order to extend the results of Theorem 1, suppose that a pair of trees (t_1, t_2) has at least one common chord e_c. If there exists minimum subgraph G_c of G satisfying

(i) $G_c \ni e_c$

and (ii) $t_1 \cap G_c$ and $t_2 \cap G_c$ are edge-disjoint trees in G_c, then G_c is said to be the K-subgraph of (t_1, t_2) with respect to common chord e_c. In terms of this definition, we can prove the following theorem:

Theorem 2: Suppose that a pair of trees (t_A, t_B) has common chords $\{e_{ci}\}$, $(i = 1,2,\ldots,m)$. Then (t_A, t_B) is maximally distinct if and only if there exists one K-subgraph with respect to each common chord e_{ci}.

Proof: In order to prove the necessity, we will show a procedure for generating K-subgraph G_{ci} with an arbitrary common chord e_{ci}. Let P_1 be the path in T_A connecting the terminal vertices of e_{ci}. If an edge $e_x \in P_1$ is in T_B, there exists a tree $t'_A = (e_{ci}/e_x)t_A$ satisfying

$$D(t'_A, t_B) = 1 + D(t_A, t_B).$$

This means that (t_A, t_B) is not maximally distinct under the assumption. Thus we can conclude that no edge in P_1 belongs to t_B. Let P_2 be the union of all the paths in t_B connecting terminal vertices of every edge in P_1. In the same reason mentioned above, it can be shown that no edge in P_2 belongs to t_A. Applying this procedure, we can obtain a sequence of sets of edges $\{P_1, P_2, P_3, \ldots\}$ which contains no edge belonging simultaneously to both t_A and t_B. Because G is finite, we can find the minimum number n satisfying

$$e_{ci} \cup P_1 \cup P_2 \cup \ldots\ldots \cup P_n \supseteq P_{n+1}.$$

Then it is easily shown that the subgraph $G_{ci} = e_{ci} \cup P_1 \cup P_2 \cup \ldots \cup P_n$ is the K-subgraph of (t_A, t_B) with respect to common chord e_{ci}.

In order to prove the sufficiency, let G_{ci} be the K-subgraph with respect to common chord e_{ci} $(i = 1,2,\ldots,m)$. Now we define the principal subgraph G_o as

$$G_o = \bigcup_{i=1}^{m} G_{ci}.$$

Suppose that G_o consists of p maximal connected components. Then it is easily shown that $t_A \cap G_o$ and $t_B \cap G_o$ form edge-disjoint foresets. Since G_o contains all the common chords, we obtain

$$m = b_o - 2(v_o - p) \qquad (2)$$

where b_o and v_o are the number of the edges and the vertices of G_o, respectively.

Let (t'_A, t'_B) be an arbitrary pair of trees with m' common chords. In order to show $m' \geq m$, let m" be the number of chords for (t'_A, t'_B) which are included in the principal subgraph G_o of (t_A, t_B). Then we can easily

$$m' \geq m'' \geq b_o - 2(v_o - p) = m. \qquad (3)$$

Applying (1), we obtain

$$D(t_A, t_B) \geq D(t'_A, t'_B).$$

This means that the pair of trees having the principal subgraph is maximally distinct. Q.E.D.

As a simple example, the pair of trees (t_1, t_2) shown in Fig. 2 is not maximally distinct, because there exists no K-subgraph with respect to common chord e_4.

Another example, given in Fig. 3, shows a maximally distinct pair of trees (t_A, t_B) and its K-subgraphs.

As an immdeiate corollary of Theorem 2, we can prove the uniqueness of the principal subgraph. If we assume that the pair of trees (t'_A, t'_B) is also maximally distinct, then Eq. (3) is

$$m' = m'' = b_o - 2(v_o - p) = m.$$

This means that $t'_A \cap G_o$ and $t'_B \cap G_o$ are edge-disjoint foresets in G_o of (t_A, t_B). Thus principal subgraph G'_o of (t'_A, t'_B) satisfies

$$G'_o \subseteq G_o.$$

Starting with (t'_A, t'_B), we can conclude that

$$G'_o \supseteq G_o.$$

Thus we obtain the following theorem:

Theorem 3: The principal subgraph is uniquely determined for all the pairs of maximally distinct trees.

The proof of the necessity of Theorem 3 suggests a procedure yielding a pair of maximally distinct trees starting with an arbitrary pair of trees. Consider, for example, the network shown in Fig. 4(a) with the pair of trees (t_1, t_2) satisfying

$$D(t_1, t_2) = 2.$$

The process yielding a pair of maximally distinct trees is as follows:

(1) Find path P_1 in t_1 connecting the terminal vertices of common chord e_1. since $\{e_2, e_5\} \subset P_1 \cap t_2$, wither e_2 or e_5 can replace e_1 to obtain distinct trees of higher distances. Thus we consider t_2 and $t_3 = (e_1/e_2)t_1$ as shown in Fig. 4(b).

(2) Consider the common chord e_3 of t_2 and t_3. Find path P_2 in t_2 connecting the terminal vertices of e_3. It is clear that P_2 and t_3 have no common edges . Find the paths in t_3 connecting all the terminal vertices of edges appeared in P_2. Then the path $\{e_4, e_5\}$ connecting the terminal vertices of e_8 and t_2 have the common edge e_5. Yielding distinct trees of higher distance, two exchange (e_3/e_8) and (e_8/e_5) can be applied to t_2 and t_3. Thus we obtain $t_4 = (e_3/e_8)t_2$ and $t_5 = (e_8/e_5)t_3$ as shown in Fig. 4(c).

(3) Consider the common chord e_{11}. Then it is easily shown that there exists the K-subgraph of (t_4, t_5) with respect to e_{11}.

(4) Similarly, we can easily observe that there exists the K-subgraph of (t_4, t_5) with respect to e_{12}.

(5) Find the path P_4 in t_4 in connecting the terminal vertices of common e_{13}. Since $\{e_6, e_7\} \subset P_4 \cap t_5$, wither e_6 or e_7 can be changed for e_{13} obtaining distinct trees of higher distance. Thus we obtain a pair of maximally distinct trees t_5 and $t_6 = (e_{13}/e_6)t_4$ as shown in Fig. 4(d).

APPLICATION

As an application of the maximally distinct trees, we will consider the generation of all the trees of a connected graph G. This provides a simple procedure yielding all the trees along a Hamilton circuit in the tree graph of G.

Before getting into the details, we will summarize the results [1][2][3] concerning the classification of trees. Let $\{e_1, e_2, \ldots, e_n\}$ be a subset of edges of a connected graph G. Then we can define a subset of trees

$$T\begin{pmatrix} e_1, e_2, \ldots, e_n \\ x_1, x_2, \ldots, x_n \end{pmatrix}, \quad x_i = 0 \text{ or } 1 \quad (i = 1, 2, \ldots, n)$$

as the collection of trees which contain e_i if $x_i = 1$ but do not if $x_i = 0$. Then the following theorems hold:

Theorem 4: For any subset of edges $\{e_1, e_2, \ldots, e_n\}$,

$$T\begin{pmatrix} e_1, e_2, \ldots, e_{n-1} \\ x_1, x_2, \ldots, x_{n-1} \end{pmatrix}$$

$$= T\begin{pmatrix} e_1, e_2, \ldots, e_{n-1}, e_n \\ x_1, x_2, \ldots, x_{n-1}, 1 \end{pmatrix} \cup T\begin{pmatrix} e_1, e_2, \ldots, e_{n-1}, e_n \\ x_1, x_2, \ldots, x_{n-1}, 0 \end{pmatrix}$$

and

$$T\begin{pmatrix} e_1, e_2, \ldots, e_n \\ x_1, x_2, \ldots, x_n \end{pmatrix} \cap T\begin{pmatrix} e_1, e_2, \ldots, e_n \\ x'_1, x'_2, \ldots, x'_n \end{pmatrix} = \phi$$

whenever

$$(x_1,x_2,\ldots,x_n) \neq (x'_1,x'_2,\ldots,x'_n).$$

Theorem 5: A necessary and sufficient condition for

$$T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ x_1,x_2,\ldots,x_n \end{pmatrix} \neq \phi$$

is the validity of the following (a) and (b):

(a) No subsets $\{e_{i1},e_{i2},\ldots,e_{im}\}$ of $\{e_1,e_2,\ldots,e_n\}$ satisfying $x_{ik} = 1,2,\ldots,m)$ form circuits in G.

(b) No subsets $\{e_{ij},e_{j2},\ldots,e_{jl}\}$ of $\{e_1,e_2,\ldots,e_n\}$ satisfying $x_{jk} = 0(k = 1,2,\ldots,l)$ form cutsets in G.

Theorem 6: The subsets of trees $T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ x_1,x_2,\ldots,x_n \end{pmatrix}$ are realizable if they are non-empty.

Theorem 7: If $T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ x_1,x_2,\ldots,x_n \end{pmatrix}$ and $T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ y_1,y_2,\ldots,y_n \end{pmatrix}$ are non-empty, and if $(x_1,x_2,\ldots,x_n)$ and $(y_1,y_2,\ldots,y_n)$ differ in only one element, then there exists at least one tree t_y in $T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ y_1,y_2,\ldots,y_n \end{pmatrix}$ for any tree t_x in $T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ x_1,x_2,\ldots,x_n \end{pmatrix}$ such that $D(t_x, t_y)=1$.

Now we turn to the application of the maximally distinct trees. Let (t_A, t_B) be a pair of maximally distinct trees in G. If we choose the subset of edges E_o defined by

$$\begin{aligned} E_o &= \{e_1,e_2,\ldots,e_n\} \\ &= \{t_A - t_B\}, \end{aligned}$$

the following theorem holds:

Theorem 8: If $E_o = \{e_1,e_2,\ldots,e_n\} = \{t_A - t_B\}$, where (t_A, t_B) is maximally distinct, then any set $T\begin{pmatrix} e_1,e_2,\ldots,e_n \\ x_1,x_2,\ldots,x_n \end{pmatrix}$ is non-empty, and there exists at least one set $T\begin{matrix} e_1,e_2,\ldots,e_n \\ y_1,y_2,\ldots,y_n \end{matrix}$ which contains only one tree.

Proof: Since no subsets of E_o form cutsets or circuits, the first assertion is an immediate consequence of Theorem 5.

In order to prove the second assertion, we assume, at first, that the number of the common chords of (t_A, t_B) is zero. For this case, it is that the set $T\begin{pmatrix} e_1, e_2, \ldots, e_n \\ 0, 0, \ldots, 0 \end{pmatrix}$ consists of t_B only. Thus we assume that (t_A, t_B) has the common chords $\{e_{c1}, e_{c2}, \ldots, e_{cm}\}$. Let G_o be the principal subgraph of G. Then E_o is divided into two components given by

$$E'_o = E_o \cap G_o = \{e_{p1}, e_{p2}, \ldots, e_{pi}\}$$

and

$$E''_o = E_o - E'_o = \{e_{q1}, e_{q2}, \ldots, e_{qj}\}.$$

Then it is easily shown that the set $T\begin{pmatrix} e_{p1}, & e_{p2}, & \ldots, & e_{pi}, & e_{q1}, & e_{q2}, & \ldots, & e_{qj} \\ 1 & ,1 & ,\ldots, & 1 & ,0 & ,0 & ,\ldots, & 0 \end{pmatrix}$ consists of the tree given by $(t_A \cap G_o) \cup (t_B - G_o)$. Q.E.D.

In order to illustrate the results of this theorem, consider the graph shown in Fig. 5. In this graph, the pair of trees (t_A, t_B) given by

$$t_A = \{e_1, e_2, e_3, e_4\}$$

and

$$t_B = \{e_4, e_5, e_6, e_7\}$$

is maximally distinct, and it is easily shown that the principal subgraph G_o is given by

$$G_o = \{e_2, e_3, e_6, e_7, e_8\}.$$

The classification of trees based on the subset of edges $E_o = \{t_A - t_B\} = \{e_1, e_2, e_3\}$ is shown in the following table: In this table, $T\begin{pmatrix} e_1, e_2, e_3 \\ 0\ ,1\ ,1 \end{pmatrix}$ consists of the tree

$$t_1 = \{e_2, e_3, e_4, e_5\}$$
$$= (t_A \cap G_o) \cup (t_B - G_o),$$

while $E'_o = \{e_2, e_3\}$ and $E''_o = \{e_1\}$.

As shown in this example, Theorem 8 provides a method of generating all the trees of a connected graph G along a Hamilton circuit in the tree graph of G. Our proposed method is a modification of [1] and stated as follows:

(1) Find a pair of maximally distinct trees (t_A, t_B) in G. Let $E_o = \{e_1, e_2, \ldots, e_n\}$ be $\{t_A - t_B\}$.

(2) Arrange all the points $(x_1,x_2,\ldots,x_n)$ ($x_i = 0$ or 1, $i = 1,2,\ldots,n$) along a Hamilton circuit on the unit cube in the n-dimensional space.

(3) Generate all the trees belonging to each subset of trees $T^{e_1,e_2,\ldots,e_n}_{x_1,x_2,\ldots,x_n}$. This can be done with the simpler graphs derived from G by short-circuiting and open-circuiting suitable edges.

(4) Start with the subset which consists of only one tree. Then the existence theorem of Hamilton circuit and Theorem 7 provide a Hamilton circuit in the tree graph.

CONCLUDING REMARKS

Fundamental properties of a pair of maximally distinct trees are discussed. A necessary and sufficient condition for maximally distinct trees is obtained in terms of theprincipal subgraph. A method of generating all the trees along the Hamilton circuit is an immediate application of the results. Relating to this application, it is mentioned that the reverse assertion of Theorem 8 can be proved. This means that the concept of maximally distinct trees can be defined in terms of the classification of trees.

REFERENCES

[1] H. Hirayama and T. Ohtsuki, "Topological Network Analysis by Digital Computer," Jour. Inst. Elect. Comm. Eng. Japan, Vol. 48, pp. 424-432, March 1965.

[2] T. Kamae, "The Existence and Generation of a Hamilton Circuit in a Tree Graph," Ph.D. Thesis, University of Illinois, Urbana, 1966.

[3] G. Kishi and Y. Kajitani, "On Hamilton Circuit in Tree Graphs," CT Monograph 97-1, Inst. Elect. Comm. Eng. Japan, November 1966.

Table 1

e_1 e_2 e_3	Trees (number n means the edge e_n)
0 1 1	t_1 = 2345
1 1 1	t_2 = 1234, t_3 = 1235
1 1 0	t_4 = 1245, t_5 = 1257, t_6 = 1247, t_7 = 1248 t_8 = 1258
1 0 0	t_9 = 1458, t_{10} = 1456, t_{11} = 1468, t_{12} = 1568 t_{13} = 1567, t_{14} = 1578, t_{15} = 1478, t_{16} = 1467
1 0 1	t_{17} = 1346, t_{18} = 1348, t_{19} = 1358, t_{20} = 1356
0 0 1	t_{21} = 3456, t_{22} = 3458
0 0 0	t_{23} = 4568, t_{24} = 4567, t_{25} = 4578
0 1 0	t_{26} = 2457, t_{27} = 2458

TOPOLOGICAL DEGREES OF FREEDOM OF ELECTRICAL NETWORKS

T. TSUCHIYA, T. OHTSUKI, Y. ISHIZAKI, H. WATANABE
Nippon Electric Co., Ltd. Kawasaki, Japan
Y. KAJITANI, G. KISHI
Tokyo Institute of Technology Tokyo, Japan

ABSTRACT

The number of topological degrees of freedom of an electrical network G is defined as the minimum number of independent variables which determine either voltage or current of every edge of G according to Kirchhoff's laws. This number is also the minimum of the rank of a subnetwork G_n plus the nullity of its complement G_m taken over all the partitionings of G into G_n and G_m.

Explicit formula for the number of topological degrees of freedom is given. Some properties of G_n^* and G_m^* which correspond to the topological degrees of freedom are studied. An algorithm for finding G_n^* and G_m^* is also presented.

I. INTRODUCTION

In the wellknown cutset or loop analysis of an electrical network, branch voltages or chord currents of a tree are taken as independent variables, respectively. More generally, the voltages of some edges and the currents of some other edges can be taken as the independent variables if they determine one of voltage and current of every edge of the network by Kirchhoff's laws. This method is known as the mixed (diakoptical [1]) analysis. The number of independent variables in the mixed analysis is called hybrid rank.

The set of independent variables are determined in accordance with a certain partitioning of the edges of the network into two parts [2][3] , and the hybrid rank of a given graph depends only upon each partitioning.

In some networks the hybrid rank can be made less than either rank or nullity, which are the number of independent variables in cutset and loop analysis, respectively. One of our main motives is to find the minimum hybrid rank, whose existence is assured from the nature of this problem. This concept may be called the topological degrees of freedom, since it depends purely upon the topology of the network.

The purpose of this paper is to present (1) an explicit formula giving the number of topological degrees of freedom and (2) an algorithm for determining a partition of a network corresponding to it.

The proofs of the lemmas and the theorems are omitted because of the lack of spaces.

II. PARTITIONING OF A GRAPH

In the mixed analysis, the set E of the edges of a given graph G is partitioned into two subsets E_n and E_m, and two new graphs $G_n = G\{E_m; \phi\}$ and $G_m = G\{\phi; E_n\}$ are considered, where $G\{A;B\}$ means the graph obtained from G by deleting the edges of A and contracting the edges of B. Clearly, independent variables for this case are the voltages of the branches of a forest T_n of G_n and the currents of the chords of a cotree CT_m of G_m. Therefore, the hybrid rank r_H, the number of independent variables of the mixed analysis, is given by

$$r_H = r(G_n) + \mu(G_m),$$

where $r(G_a)$ and $\mu(G_a)$ are the rank and the nullity of a graph G_a, respectively.

Definition 1

$$d = \min(r_H)$$

is called the number of (topological) degrees of freedom or the minimum hybrid rank of the graph, where the minimum is taken over all possible partitionings of the set of edges of the graph.

Lemma 1

$$r(G_m)+r(G_n)= r(G)$$
$$\mu(G_m)+\mu(G_n)= \mu(G)$$

As the immediate consequence of this lemma we have the following.

Theorem 1

$$\begin{aligned} d&=\min\left[\mu(G)-\left\{\mu(G_n)-r(G_n)\right\}\right] \\ &=\min\left[r(G)-\left\{r(G_m)-\mu(G_m)\right\}\right] \end{aligned}$$

For the convenience of further arguments, the following notations will be introduced.

G_n^* (G_m^*) : a $G_n(G_m)$ which corresponds to a partition giving d.

$\bar{G}_n^*$ $(\bar{G}_m^*)$: a G_n^* (G_m^*) which has the maximum rank (nullity) among all possible $G_n^*(G_m^*)$'s

$\underline{G}_n^*$ $(\underline{G}_m^*)$: a $G_n^*(G_m^*)$ which has the minimum rank (nullity) among all possible $G_n^*(G_m^*)$'s

E_n^* (E_m^*) : the set of edges of $G_n^*(G_m^*)$

$\underline{E}_n^*$ $(\underline{E}_m^*)$: the set of edges of $\underline{G}_n^*(\underline{G}_m^*)$

$\bar{E}_n^*$ $(\bar{E}_m^*)$: the set of edges of $\bar{G}_n^*(\bar{G}_m^*)$

Lemma 2

(1) In each non-separable component G_{ni}^* of G_n^*

$$\mu(G_{ni}^*)\geq r(G_{ni}^*)$$

(2) In each non-separable component $\underline{G}_{ni}^*$ of $\underline{G}_n^*$

$$\mu(\underline{G}_{ni}^*)> r(\underline{G}_{ni}^*)$$

Theorem 2

(1) Both $\underline{G}_n^*$ and $\bar{G}_n^*$ are unique in a given G.

(2) For an arbitrary G_n^* we have

$$\underline{E}_n^*\subset E_n^*\subset \bar{E}_n^*.$$

Corollary 1

(1) $\bar{E}_m^* = E \setminus \underline{E}_n^*$

(2) $\underline{E}_m^* = E \setminus \bar{E}_n^*$

Corollary 2

The following four statements are equivalent:

(1) G_n^* can be determined uniquely.

(2) G_m^* can be determined uniquely.

(3) $\underline{E}_n^* = \bar{E}_n^*$

(4) $\underline{E}_m^* = \bar{E}_m^*$

III. EXTREMAL TREE AND MAXIMALLY DISTINCT TREE PAIR

In this section we shall introduce a new concept, extremal tree, which will play an important role in finding the topologcal degrees of freedom. This concept coincides essentially with that of maximally distinct tree pair which is proposed by two of the authors [4].

Lemma 3

Let b and c, respectively, be the number of common branches and common chords in a tree-pair of a connected graph G, then we have

$$r(G)-b = \mu(G)-c .$$

Lemma 4

Let $r_s(\mu_o)$ be the number of independent cutsets (circuits) of G which consist only of branches (chords) of a tree T.
And let T' be a tree which is most distant from T . Then for the tree-pair (T-T'), we have

$$r_s = b$$
$$= \text{rank of } G\{\phi ; E \setminus T\}$$

and

$$\mu_o = c$$
$$= \text{nullity of } G\{T; \phi\}.$$

We now are in a position to introduce the following definition.

Definition 2

A tree (cotree) $T^*(CT^*)$ which has the minimum $r_s(\mu_o)$ in all trees in G is called an extremal tree (cotree).

Theorem 3

The following three statements are equivalent:

(1) T is an extremal tree.

(2) The complement of T is an extremal cotree.

(3) T and its most distant tree T' constitute a maximally distinct tree pair.

IV. ALGORITHM FOR GENERATING AN EXTREMAL TREE

Let T be a tree of G and CT be its complement. By $E\mu$ $(=T\mu \cup CT\mu)$ we denote the minimal set of edges satisfying the following conditions:

(1) $T\mu \subset T$ and $CT\mu \subset CT$

(2) $T\mu$ contains all edges in a cutset which is composed only of the branches of T.

(3) Every edge of $CT\mu$ belongs to at least one of the fundamental cutsets defined by the branches of T.

(4) In $G\{E\mu;\phi\}$, there exists no cutset that is composed only of the branches of T.

Dually, we denote by $E\nu$ $(=T\nu \cup CT\nu)$ the minimal set of edges satisfying the following conditions:

(1') $T\mu \subset T$ and $CT\mu \subset CT$.

(2') CT contains all edges in a circuit which is composed only of the chords of T.

(3') Every edge of T belongs to at least one of the fundamental circuits defined by the edges of CT.

(4') In $G\{\phi;E\nu\}$, there exists no cutset that is composed only of the chords of T.

For each tree T we can define two graphs $G\mu(T)$ and $G\nu(T)$ such that:

Definition 3

$$G\mu(T) = G\{\phi\,;(E\setminus E\mu)\},$$

$$G\nu(T) = G\{(E\setminus E\nu);\phi\}.$$

Lemma 5

Let $t\mu$ (t_ν) and $l\mu$ (l_ν) be the number of edges in $T\mu(T_\nu)$ and $CT\mu(CT_\nu)$, respectively. And let $r_s(\mu_o)$ be the rank(nullity) of $G\{\phi;CT\}$ $(G\{T;\phi\})$.

Then we have

$$l\mu + r_s \geq t\mu \qquad (t\nu + \mu_o \geq l\nu),$$

where the equality holds if and only if there exists no circuit (cutset) composed only of the edges of $CT\mu(T_\nu)$.

As is clear from definition 3, we can construct the graph $G\mu$ corresponding to a tree T by the following algorithm,

Algorithm 1

Step 0 : Let k=1 and $G^{(o)} = G$. Go to Step 1.

Step 1 : Let $T^{(k)}_{\mu}$ be the set of edges in a cutset of $G^{(k-1)}$ which is composed only of the branches of T. If $T^{(k)}_{\mu} = \phi$, then $G_{\mu}(T)$ has been constructed by

$$G\mu(T)=G\{\phi \; ; \text{ set of edges in } G^{(k-1)}\} .$$

Else go to Step 2.

Step 2 : Let $CT^{(k)}_{\mu}$ be the set of chords of T such that each edge of $CT^{(k)}_{\mu}$ belongs to at least one of the fundamental cutsets of $G^{(k-1)}$ defined by the edges of $T^{(k)}_{\mu}$. Go the Step 3.

Step 3 : Define a new graph $G^{(k)}$ as follows:

$$G^{(k)} = G^{(k-1)}\{T^{(k)}_{\mu} \cup CT^{(k)}_{\mu} \; ; \phi\} .$$

Go to Step 1, letting k=k+1.

The sets of edges $CT^{(k)}_{\mu}$ and $T^{(k)}_{\mu}$ defined in the procedure above have the following properties.

Lemma 6

(1) Each fundamental circuit of G defined by an edge of $CT^{(k)}_{\mu}$ is composed only of the edges in $T^{(k+p)}_{\mu} (p \geq 0)$, and contains at least one edge of $T^{(k)}_{\mu}$.

(2) For each edge of $T^{(k)}_{\mu}$ $(k \geq 2)$, there exists a circuit which contains no branch of T except this edge. In addition, this circuit contains at least one edge of $CT^{(k-p)}_{\mu} (p \geq 1)$.

Let r_{s1} be the number of independent cutsets composed only of the branches of a tree T_1 of G, and assume that some chords of T_1 form a circuit in $G\mu(T_1)$, i.e. $l\mu_1 + r_{s1} > t\mu_1$. Then we can generate, using the properties stated in Lemma 6, a new tree T_2 such that $r_{s2} = r_{s1} - 1$ by the following procedure:

Algorithm 2

```
Step 0 :  In $G\mu$ ($T_1$), take a circuit C composed only of the chords
          of $T_1$.
          Go to Step 1.

Step 1 :  Let $e_c$ be an edge of C such that it belongs to the set
          $CT_\mu^{(k)}$ and that no other edge of C belongs the set $CT_\mu^{(k-p)}$ ($p\geq 1$).
          Go to Step 2.

Step 2 :  Let $e_b$ be an edge of $T_\mu^{(k)}$ such that it belongs to the
          fundamental circuit defined by $e_c$. (Existence of $e_b$ is
          assured by Lemma 6 - (1).). Generate new tree $T_2$
          from $T_1$ by replacing $e_b$ with $e_c$.
          If k=1, then $T_2$ is a tree which satisfies $r_{s2}=r_{s1}-1$.
          Else go to Step 3.

Step 3 :  Take the circuit C created by Step 2, i.e. C is composed
          only of the chords of $T_2$ and contains $e_b$. Go to Step 1,
          letting $T_1=T_2$.
```

This process comes to an end in finite steps, since Lemma 6 - (2) assures that the superfix k decreases whenever the process returns from Step 3 to Step 1.

The dual of Algorithm 1,2 with respect to G will be omitted in this paper. Summarizing the results in this section we have the theorem.

Theorem 4

The following five statements are equivalent to each statement of Thorem 3.

(1) $G\mu$ (T) contains no circuit which is composed only of the chords of T.

(1') $G\nu$ (T) contains no cutset which is composed only of the branches of T.

(2) In $G\mu(T)$, $l\mu+r_s=t\mu$.

(2') In $G\nu$ (T), $t\nu+\mu_o=l\nu$.

(3) $G\mu(T)$ and $G\nu(T)$ have no common edge.

Theorem 4 - (1) shows that we can obtain an extremal tree, starting with an arbitrary tree, by successive applications of Algorithm 1 and 2.

$G\mu(T^*)$ and $G\nu(T^*)$ defined for an extremal tree T^* coincide with G_1 and G_2, respectively, in [4].

V. RELATION BETWEEN THE EXTREMAL TREE AND THE NUMBER OF THE TOPOLOGICAL DEGREES OF FREEDOM

In order to present an explicit formula for the number of topological degrees of freedom and to find $\underline{G}_n^*$ and $\underline{G}_m^*$, some properties of G_n^* (G_m^*) will be investigated.

In the following we denote μ_o and r_s for the extremal tree by μ_o^* and r_s^*, respectively.

Lemma 7

(1) In each non-separable component of G_n^*, there exists a tree which does not contain cutsets composed only of branches of the tree.

(2) $\mu(G_n^*) - r(G_n^*) = \mu_o^*$

(3) Let T^* be an arbitrary extremal tree, and G_{ni}^* be a non-separable component of G_n^*, then $G_{ni}^* \cap T^*$ is a tree of G_{ni}^*.

(4) If a chord of an extremal tree T^* does not belong to E_n^*, then the fundamental circuit associated with this chord contains at least one branch of T^* not belonging to E_n^*.

The following lemma is the dual of Lemma 7.

Lemma 7'

(1) In each non-separable component of G_m^*, there exists cotree which does not contain circuits composed only of the chords.

(2) $r(G_m^*) - \mu(G_m^*) = r_s^*$

(3) Let T^* be an arbitrary extremal tree and G_{mi}^* be a non-separable component of G_m^*, then $G_{mi}^* \cap T^*$ is a tree of G_{mi}^*.

(4) If a branch of an extremal tree T^* does not belong to E_m^*, then the fundamental cutset associated with this branch contains at least one chord of T^* not belonging to E_m^*.

From the above lemmas, explicit expressions for the number of the topological degrees of freedom are obtained.

Theorem 5

$$d = r(G) - r_s^* = \mu(G) - \mu_o^*$$

Corollary

(1) $d = r(G) - b^* = \mu(G) - c^*$,

where b^* and c^* are the number of the common branches and the common chords, respectively, of a maximally distinct tree pair.

(2) $d = \max\ (r(G) - r_s) = \max\ (\mu(G) - \mu_o)$,

where the maximum is taken over all trees of G.

In order to find a set of variables corresponding to the topological degrees of freedom, we have to determine a partitioning of G into G_n^* and G_m^*. The following theorem says G_n^* and G_m^* can be constructed by successive applications of Algorithms 1 and 2.

Theorem 6

For an arbitrary extremal tree T^*,

$$\underline{G}_n^* = G\nu(T^*),\ \underline{G}_m^* = G\mu(T^*).$$

Corollary

$G\nu(T^*)$ and $G\mu(T^*)$ are uniquely determined independent of the choice of an extremal tree T^*.

The following theorems are also an interesting results on $\underline{G}_n^*$ and $\underline{G}_m^*$.

Theorem 7

(1) $En^* = \bigcup$ (the set of edges forming circuits in

$G\{T, \phi\}$),

where the union is taken over all the extremal trees of G.

(2) $E_n^* = \bigcup$(the set of the common chords of a tree pair),

where the union is taken over all the maximally distinct tree pairs.

Theorem 7'

(1) $Em^* = \bigcup$ (the set of edges forming cutsets in

$G\{\phi, T\}$),

where the union is taken over all the extremal trees of G.

(2) $Em^* = \bigcup$(the set of common branches of a tree pair),

where the union is taken over all the maximally distinct tree pairs.

VI. AN ILLUSTRATIVE EXAMPLE

Taking a graph shown in Fig. 1 as an example, we illustrate how we can construct $\underline{G}_m^*$ and $\underline{G}_n^*$ according to Algorithms 1 and 2.

(1) Take an arbitrary tree T_1 in a given graph G. T_1 is shown by bold lines in Fig. 1(a).

(2) Using Algorithm 1, define the edges of $T_\mu^{(k)}$ and $CT_\mu^{(k)}$, successively, until $G\mu(T_1)$ has been constructed. The number k for each edge of $G\mu(T_1)$ is shown in parentheses.

(3) It can be seen from Theorem 4 – (1) that T_1 is not an extremal tree. Take a circuit C, which is shown by dotted lines. (Algorithm 2 – Step 0)

(4) Let an arbitrary edge of C be e_c, since the same number 2 is attached to every edge of C. (Algorithm 2 – Step 1)

(5) Take as e_b a branch numbered k=2 (in this case any edge) in the fundamental circuit defined by e_c. Then we have a new tree T_2, which is shown in Fig. 1(b), by replacing e_b with e_c. (Algorithm 2 – Step 2)

(6) In Fig. 1(b), the dotted line represent the new circuit composed only of the chords of the new tree T_2. (Algorithm 2 – Step 3)

(7) Repeating Algorithm 2, we have a tree T_3 as shown in Fig. 1(c). T_3 is an extremal tree. (Theorem 4 – (1))

(8) Since T_3 is an extremal tree, the set of the numbered edges in Fig. 1(c) is $\underline{E}_m^*$. (Theorem 6)

(9) Applying the dual of Algorithm 1 with respect to T_3, we have the set $\underline{E}_n^*$.

$\underline{G}_m^*$ and $\underline{G}_n^*$ thus constructed are shown in Fig. 2(a) and (b), respectively.
$\overline{G}_n^*$ and $\overline{G}_m^*$, shown in Fig. 2(c) and (d), respectively, can be constructed by Corollary 1.

In this example,

$$r(G) = \mu(G) = 9,$$

$$r(\underline{G}_n^*) = \mu(\underline{G}_m^*) = 2, \quad r(\overline{G}_n^*) = \mu(\overline{G}_m^*) = 5.$$

So we can see the number of topological degrees of freedom is less than either rank (=9) or nullity (=9).

VII. CONCLUSIONS

A new concept the topological degrees of freedom has been introduced in giving the minimum number of independent variables in the mixed analysis of electrical networks. Important properties of topological degrees of freedom d have been presented. In particular, the relations between topological degrees of freedom d and newly defind extremal tree enable us to determind d explicitly from a given graph. An algorithm of finding the extremal tree and actual partitionings giving d from the graph has been also given in detail.

It should be noted that the topological degrees of freedom is not only the essential quantity of topology of a graph and the index of the complexity of a graph, but also it gives the minimum number of independent equations of electrical network analysis or the minimum order of matrix to be inverted. In this sence, the topological degrees of freedom and related results, presented in this paper, are important both in graph theory and for time saving computation of network analysis. Detailed studies of latter will be presented in the nearest future.

ACKNOWLEDGEMENT

The authors wish to express their hearty appreciations to Prof. Masao Iri of University of Tokyo for his stimulative discussions and helpful suggestitions throughout this work.

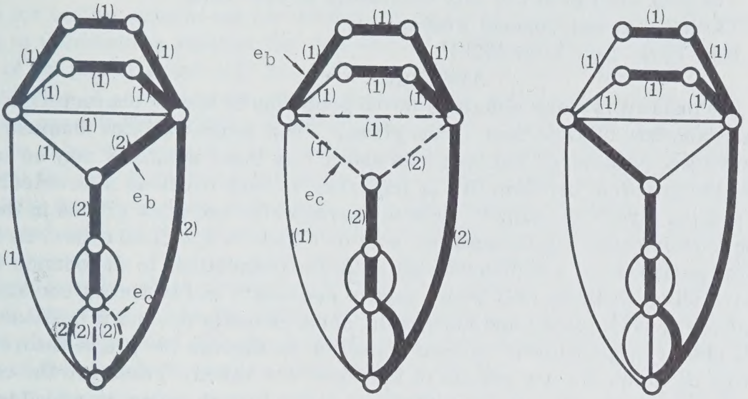
REFERENCES

〔1〕 G. Kron, "Diakoptics", The Electrical Journal, London, 1957-1959.

〔2〕 S. Amari, "Topological Foundations of Kron's Tearing of Electric Networks", RAAC Memoirs vol. 3 1962.

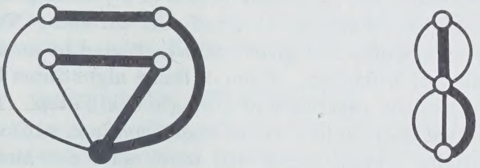
〔3〕 R.K. Brayton and J.K. Moser, "A Theory of Nonlinear Networks", Quart. Appl. Math., vol. 22, April, 1964.

〔4〕 G. Kishi, Y. Kajitani, "On Maximally Distinct Trees", Fifth Annual Allerton Conference on Circuit and System Theory, 1967.

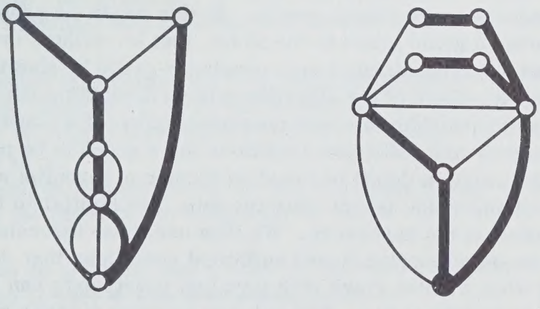


(a) T_1 (b) T_2 (c) T_3

Fig. 1 Generation of an external tree



(a) $\underline{G}_m{}^*$; $\mu(\underline{G}_m{}^*)=2$ (b) $\underline{G}_n{}^*$; $r(\underline{G}_n{}^*)=2$



(c) $\bar{G}_n{}^*$; $r(\bar{G}_n{}^*)=5$ (d) $\bar{G}_m{}^*$; $\mu(\bar{G}_m{}^*)=5$

Fig. 2 Partitionings into $G_n{}^*$ and $G_m{}^*$

TWO NEW CHARACTERIZATIONS OF PLANAR GRAPHS[†]

LOUIS WEINBERG
The City College of the City University of New York
139th Street and Convent Avenue
New York, New York 10031

ABSTRACT

We present two new solutions to the problems of how to characterize planar graphs and how to draw them in the plane. First graphs without terminal constraints are considered and then it is shown how these solutions may be adapted to solve the practical problem that is important in such areas as microelectronics and printed circuitry, namely, how to characterize and draw graphs in the plane when certain nodes are designated, with or without a specified order, as terminals (on the periphery of a semiconductor chip) for connections to the outside world. After stating the three well known sets of necessary and sufficient conditions on a graph for it to be planar and mentioning some recently published methods for drawing a planar graph without crossed branches, we discuss the graph-theoretic properties on which the new results of the paper are based. These are the concept of the connectivity of a graph and the fact that a branch cannot be added to a plane graph without causing a crossover unless the nodes of the branch are in a common mesh. By use of a third concept of the unique set of meshes of a planar triply connected graph or of any plane graph we convert the results so that they apply to graphs with terminal constraints. Then we state two new theorems, each of which gives a necessary and sufficient characterization of a planar graph, one theorem in terms of circuits and the second in terms of star cut sets. Two simple algorithms based on these theorems are given and illustrated by an example of a nontrivial graph containing 21 branches. Each of these algorithms has the attractive feature that it consists of the repetition of a single basic step. These theorems are then extended so that they yield a set of necessary and sufficient conditions and an algorithm for realizing a plane graph with terminal constraints. Finally, in an Appendix we present an unpublished algorithm formulated by A.J. Goldstein for drawing a graph in the plane.

INTRODUCTION

We are concerned with planar graphs. In this paper we present two new algorithms for drawing a given graph in the plane, that is, without crossed branches, if it is planar, and for recognizing it as a nonplanar graph by showing the impossibility of so drawing it. Each of the algorithms is so direct that the enuciated theorem embodying it can be considered a new characterization of a planar graph, that is, a new set of necessary and sufficient conditions for a graph to be planar. The algorithm that is discussed in detail is based on the use of circuits; a second theorem and algorithm which make use of star cut sets, are similar to the first so that a detailed discussion is not necessary. We then use these theorems as necessary conditions in the set of necessary and sufficient conditions that solves the problem of determining when a given graph with terminal constraints can be drawn in the plane without crossed branches. The precise meaning of these constraints is explained later in the text.

[†] The research reported in this paper was supported by the NSF under Grant GK 1093

There are at least three characterizations of planar graphs. None of these is constructive in the sense that it yields an efficient procedure as a planarity test. Even if a graph is known to be planar, the characterizations do not indicate how to draw it in the plane. There are, however, recently developed algorithms for testing graphs and for drawing them in the plane. Though the procedure due to Goldstein is simpler than the others, it appears that the basic simplicity of the planar graph still has not been utilized in these algorithms. What is desirable is a simple rule, easily remembered and easily applied, that could be used repeatedly to yield a drawing for a planar graph. The simplicity would also be valuable for saving calculation time in a computer implementation.

To present the new algorithms and prove the basic theorems on which they are based, we first introduce some definitions. We also indicate the previous work on the problem and some of the salient properties of planar graphs in order to place the problem in context. Thus the apparatus we develop and the discussion we give are not inconsiderable. It is pleasant to inform the reader at the outset, however, that the first algorithm is like the simple intuitive process used by most college sophomores when presented with the problem of drawing a graph in the plane; thus the wisdom of the unsophisticated is again shown to be deep. The college sophomore generally starts with a circuit and in each step adds a branch or a path to form new circuits without crossings--until he gets stuck. If he finally decides nothing more can be done, he is never certain whether this is so because the graph is truly nonplanar or that another order of steps might have led to a drawing in the plane. At this point we bring in sophistication to resolve the sophomore's dilemma; this includes the concept of the connectivity of a graph and its consequences for planarity, and secondly, the simple property that requires the two nodes of a branch to be on a common mesh in order that it be possible to add the branch without introducing crossings.

DEFINITIONS AND CONCEPTS

We assume the reader is familiar with the basic concepts of linear graphs or can consult the books [1,2,3] on graph theory for definitions of the terms used in the paper. However, we do present some definitions necessary to make the discussion immediately intelligible.

A graph is simple if it has at most one branch joining each pair of nodes; in electrical terminology, the graph has no branches in parallel. Graphs in which any number of parallel branches may be connected between a pair of nodes are called multigraphs.

A graph G is said to be k-connected or k-tuply connected if it contains at least k+1 nodes and is not transformed to an unconnected graph by the deletion of less than k nodes. If G is k-tuply connected but not (k+1)-tuply connected, its connectivity is defined as k.

Related to the concept of connectivity is that of k-separators. A graph G is k-separated if it is the union of two subgraphs H and K with the following properties:

a) H and K have no common edges, that is, they are complements.

b) H and K have not more than k common nodes.

c) Each of the subgraphs H and K has a node not belonging to the other.

We call the pair {H,K} a k-separator of G. A graph which is not k-separated is called (k+1)-connected. Thus a graph that is not 2-separated is a triply connected graph.

The number and types of paths in a graph constitute a basic property for determining the connectivity of a connected graph, and, more specifically, the pair of common nodes of a 2-separator and the common node of a 1-separator. A necesary and sufficient condition for a simple graph to be k-connected is that any pair of nodes be connected by at least k distinct paths, that is, paths with no nodes in common except the end nodes. Thus a simple triply connected graph must have at least three such paths between each pair of nodes. If a graph is not triply connected because each path between a pair of nodes goes through at least one of two nodes, but all paths do not go through only one of these nodes, then these two nodes are the common nodes of the components of a 2-separator. If all paths between a pair of nodes have a node in common, then this node is an articulation or separation node, and the graph can be separated into its 1-separator components. An unconnected graph or a 1-separated graph is said to be separable; all other graphs are nonseparable.

A tree is a connected graph without circuits. In a graph of connectivity 2 each pair of branches is a part of a circuit.

An important class of graphs is the series-parallel graph; to describe a test for such a graph we give some preliminary definitions. A two-terminal graph is a connected multigraph in which two nodes marked u and v are called the first terminal and the second terminal, respectively. The series connection of the two-terminal graphs G_1 and G_2 is the graph obtained by identifying the nodes v_1 and u_2; the graphs are said to be in series. The parallel connection of G_1 and G_2 is obtained by identifying the nodes u_1 and u_2 and the nodes v_1 and v_2; the graphs are said to be in parallel. Thus the two components in a 2-separator are in parallel at the common nodes. The definition of a series-parallel graph may now be given recursively: a single branch is a series-parallel graph; two series-parallel graphs in series or in parallel form a series-parallel graph.

It should be noted that in the definition of a series-parallel graph that is generally given a pair of terminals must be specified with respect to which the graph is series -parallel. A graph may be series-parallel with respect to some pairs of nodes but not with respect to other pairs; for example, the graph formed from K_4, the complete graph on four nodes, by removing any branch is not series-parallel with respect to the two nodes previously incident to the removed branch, but is series-parallel with respect to any other two nodes. However, we may define a series-parallel graph in a different way without specification of terminals; namely, a graph is series-parallel if it is a series-parallel two-terminal graph with respect to any pair of adjacent nodes, that is, any pair of nodes joined by a branch.

To determine whether a graph is series-parallel we merely repeat a basic step: replace each set of branches in series by a single branch and replace each set of parallel branches by a single branch. If the repetition of this step until it is no longer possible leads to a graph consisting of a single branch, the given graph is series-parallel; otherwise it is not. It is clear that in combining two branches in series we merely delete one branch and identify its two nodes; in combining two branches in parallel we delete one branch.

The above concepts can now be applied to planar graphs, but again we first need some additional definitions. A planar graph is a graph that can be drawn in the plane without crossed branches; such a drawing is called a plane graph. The branches of a plane graph form a set of adjoining polygons which divide the plane into regions called faces; thus a face is a region bounded by a circuit, where the region contains no other circuits. This particular type of circuit is designated as a mesh. Thus a mesh is an ordered set of branches (or nodes) that encircle the corresponding face. There are F meshes (or faces) including the outer mesh (or corresponding infinite face) in a plane graph; the number of meshes is related to the number of branches b and the number of nodes n in a connected plane graph by Euler's formula

$$F + n - b = 2$$

Any plane graph has a unique set of meshes, but there are in general many plane representations for a planar graph. This is not true, however, for a triply connected graph; this graph has a unique plane representation, where we consider a mirror image of the plane representation as the same representation, in other words, the ordered set of nodes representing each mesh is considered as a clockwise order to obtain one representation and as a counterclockwise order to obtain the mirror-image graph. For a graph of connectivity 2 all representations, that is,, all the different sets of meshes, may be obtained by permutation of parallel subgraphs at the two common nodes of each 2-separator or by rotation of one subgraph that is in parallel at these nodes. The precise procedure for generating all plane representations without redundancy is given by the Mac Lane characterization discussed briefly below.

It is intuitively obvious (and may be proved by use of the Jordan curve theorem) that a branch (b, c) may be added between nodes b and c of a plane graph without crossing other branches if and only if nodes b and c are on a common mesh. Thus if a plane graph G is a triply connected graph and b and c are not on a common mesh, the addition of branch (b, c) to G yields a nonplanar graph. However, if G is not triply connected, nodes b and c may be made to lie on a common mesh by permuting any subgraphs that are in parallel at the common nodes of a 2-separator, or by rotating a subgraph connected between these two nodes, or by rotating a subgraph at an articulation node. The permutation may have to be applied twice, once at each of two pairs of nodes, where the two pairs of nodes must lie on a common mesh. These pairs of nodes may be found by determining the paths between nodes b and c. Thus, for example, if we wish to add a branch between x and y in Fig. 1a, we find that the pair of nodes e and b and the pair c and d are nodes of 2-separators on paths between x and y and lie on the common mesh e c d b e; two permutations of subgraphs then yield the new set of meshes in Fig. 1b so that branch (x, y) may now be added without a crossover. However, in Fig. 2, though each pair of nodes c and d and pair e and f are common nodes of a 2-separator, as made evident by considering the paths between x and y, they do not lie on a common mesh, and thus it is not possible to place x and y in a common mesh by permutations of subgraphs.

If the graph has connectivity 1, we may have to rotate a node that is not on a mesh about an articulation node and sometimes, in addition, permute subgraphs about the common nodes of a 2-separator; the articulation node and the pair of nodes of the 2-separator must be on a common mesh. Such a case is shown in Fig. 3a, where the result after the rotation about node b and permutation of subgraphs about nodes c and d is shown in Fig. 3b. As will become clear later, the scheme we use for indicating the meshes of a node that is not actually in a mesh assumes that this rotation can be carried out. Thus each of the nodes y, z, and w in Fig. 3 is said to be on the meshes of the articulation node b; for example, each is on the outer mesh f e b d f.

A tree is clearly a planar graph, and a plane representation is found easily. A series-parallel graph is also planar; a plane representation may be found by reversing the process for testing whether a graph is series-parallel, since in any plane graph each branch may be replaced by a parallel set of branches or by a series set of branches without destroying planarity.

A set of F - 1 circuits (called a fundamental set of circuits) that may serve as a basis of the vector space of circuits, modulo 2, may be found by use of a tree; each branch of the graph that is not in the tree is called a link and forms a fundamental circuit with some of the tree branches.

CHARACTERIZATION OF PLANAR GRAPHS

As mentioned previously, there are three well known sets of necessary and sufficient conditions on a planar graph. The first is due to Kuratowski, the second to Whitney, and the third to Mac Lane. They are:

1) A graph is planar if and only if it does not contain as a subgraph either of the two basic nonplanar graphs, namely, K_5, the complete graph on five nodes, and the complete bipartite graph $K_{3,3}$, or homeomorphs of these two graphs. (By a homeomorph is meant a graph in which one or more branches are replaced by a series connection of branches.)

2) A graph is planar if and only if it has a dual.

3) A graph is planar if and only if it contains a complete set of circuits such that no branch appears in more than two of these circuits. By a complete set of circuits we mean a basis of the vector space of circuits, modulo 2. It is clear that these circuits are any F - 1 meshes of the F meshes of any plane representation of the graph.

To test a graph for planarity there are a number of methods available [4,5,6]; recently a new method has been formulated [7]. The method of Goldstein, which is unpublished, is given in the Appendix.

Most of the methods consider the given connected graph in its entirety; the procedure due to Lempel et al, however, starts with a biconnected graph. We go further and show that a simple triply connected graph can be the starting point without any loss of generality. It is clearly not necessary to consider unconnected graphs since an unconnected graph is planar if and only if its connected components are planar. If the connected graph is separable, we may divide it into its separable components, and, according to a theorem of Whitney, the graph is planar if and only if its components are planar. Of course, if one of the components is a tree, it is not necessary to divide it into its components; the preliminary testing procedure should therefore detect trees and keep them intact as a single component. Thus the preceding establishes that only biconnected components need be considered. Furthermore, if the biconnected graph is a series-parallel graph, it may be drawn without use of the basic algorithm.

Moreover, there is a structural characterization of planar graphs due to Mac Lane [8] which shows that we need consider only simple triply connected graphs. If the graph G is not triply connected, we find a 2-separator {H, K} of G. We then connect a branch between the common nodes of H and a branch between the common nodes of K. We then repeat the step to find a 2-separator on each of these two augumented 2-separator components. This process is repeated as often as necessary until we obtain triply connected components. Of course, it simplifies the procedure if we test the components for series- parallel character, since it is not necessary to further reduce a series-parallel component. The theorem given by Mac Lane shows that the original graph is planar if and only if its components are planar. Finally, if we eliminate parallel branches in a triply connected component, we see that we need test only simple triply connected graphs.

Though the above simplification is available to the computer programmer, we present our algorithm in the next section so that it applies to any connected graph not merely to a triply connected graph. However, even if the user doesn't avail himself of Mac Lane's characterization in its full generality, an obvious simplification of a given graph is to replace each set of series branches and each set of parallel branches by a single branch.

FIRST BASIC THEOREM, ITS PROOF, AND ALGORITHM

The basic theorem may be stated in a variety of forms each of which leads to a slightly different algorithm. Of course, what is desirable is an algorithm that terminates as early as possible when applied to a nonplanar graph and still uses little computational time to draw a plane representation when the graph is planar.

Suppose we are given a connected graph which we wish to test for planarity. As mentioned above, we can simplify the problem by replacing each set of series branches by a single branch and each set of parallel branches by a single branch to obtain graph G. The theorem and the algorithm still apply even if this simplification is not used. We number the nodes of G arbitrarily 1,2, ... ,n. We now draw any tree of G and list the links in any order $L_1, L_2, \ldots, L_{F-1}$. Each link is of course defined by its two nodes; thus $L_1=(b,g)$, $L_2=(c,f), \ldots$, where b, g, c, f, ..., are node numbers. We now repeat a basic step that draws a set of subgraphs $G_1, G_2, \ldots, G_{F-1}=G$, where G_1 consists of the tree augmented by link L_1, G_2 consists of G_1 augmented by link L_2, and in general G_r consists of G_{r-1} augmented by L_r.

When we add L_1, a fundamental circuit is formed. We label this circuit as mesh M_1 and the outer mesh as M_2. We then label each node with the meshes on which it lies; in a practical implementation we may check only the meshes of the link nodes of the next link to be added. If a node is on a 1-separator component and is not part of a mesh, it is labelled with the meshes of the node that is both an articulation node of this 1-separator and is part of a mesh. It is clear that such nodes can be made to lie in these meshes by a rotation about the articulation node. Now when we add L_2, another mesh is formed which is labelled M_3. If the nodes of L_2 lie on M_1 we have a choice of placing L_2 in M_1 or M_2. Whenever we have such a choice, we may choose the mesh arbitrarily since the choice makes no difference in the successful working of the algorithm. It may occur that a node not on a mesh is part of two different 1-separators, where each articulation node is on a mesh. The node is labelled with the meshes of both articulation nodes since it may be rotated about either node.

We now can state the basic theorem on which the first algorithm is based:

Theorem 1. A graph G is planar if and only if each of its subgraphs G_r, where $r=1,2,\ldots,F-2$, is a plane graph and the two nodes of L_{r+1}, the next link to be added, lie on a common mesh or can be made to lie on a common mesh by rotation of a subgraph about an articulation node or about the common nodes of a 2-separator, or by permutation of subgraphs connected in parallel between the two common nodes of a 2-separator of G_r.

Proof. To show necessity we assume G is planar. Now suppose that the hypothesis on the subgraphs is not true for a subgraph G_r. Since all the possible sets of meshes of G_r can be generated by permutations of parallel subgraphs or by rotation of a parallel subgraph, we cannot find a plane representation such that the nodes of L_{r+1} lie on a common mesh. Thus L_{r+1} cannot be added without crossing a branch of every plane representation of G_r, and G_{r+1} is nonplanar. This leads to the contradiction that G is nonplanar.

For showing sufficiency we assume the hypothesis is true. Thus at each step we can add a link without causing a crossover. We thus obtain the plane graph G. This completes the proof.

As a variation on the theorem, after adding a link, we can inspect the resulting subgraph to determine whether each of the node-pairs of all succeeding links lie on a common mesh. This requires more examinations at each step, but often can lead to a shorter test showing that a given graph is nonplanar.

The algorithm now follows. We assume that as an initial step the connected graph is divided into its 1-separator components, where trees, however, are not decomposed further but kept intact as a single connected component. Each component may be handled separately, and if a plane representation is found for each, the articulation nodes may then be joined again. Similarly, in each component we replace each set of series and parallel branches by a single branch. It is also suggested that one use as a preliminary step the Mac Lane decomposition into components that are simple triply connected graphs or series-parallel graphs. Only the former need be tested by use of the algorithm. However, it is stressed that the algorithm as stated applies even if these steps have not been carried out, but, it should be clear, the computational labor is increased.

1) Label the nodes of the given graph G with the integers 1, 2, ..., n.

2) Choose any tree of G and draw it in the plane.

3) Arbitrarily label the links $L_1, L_2, \ldots, L_{F-1}$.

4) Now repeat the basic step of adding each of the links L_r in order to the subgraph drawn in the plane. After link L_r is drawn to give subgraph G_r, we list the meshes incident to each of the two nodes of L_{r+1}. A link may be added if its nodes are in a common mesh. If they are not, we rotate subgraphs about an articulation node, or rotate or permute subgraphs about a pair of common nodes of a 2-separator as explained previously to redraw the graph so that the link nodes are now in a common mesh. If it is impossible to do this for the link to be added, the given graph is nonplanar. If all links can be added in this way, the graph is planar and a plane representation has been drawn.

A comment about the ordering of the links should be made. It is desirable to make the subgraph already drawn in the plane a triply connected graph as early as possible; then no subsequent permutations of subgraphs will be required. To accomplish this, we can order the links so that each of the nodes of the drawn subgraph achieves a valence of at least three as early as possible; we start with nodes of valence two in the tree and then subsequently consider nodes of valence two in the resulting subgraphs. If there are no more nodes of valence two at which we can add a link, we then add links to nodes of valence one.

AN ALTERNATIVE ALGORITHM

We now discuss briefly the second algorithm which is based on cut sets rather than circuits. We use (n - 1) star cut sets, where the star cut set at node r is the set of all branches incident to node r. Precisely the same principles on rotation of 1-connected subgraphs and rotation or permutation of parallel subgraphs apply to this case. The proof of the corresponding theorem also follows easily. After numbering the nodes of the given graph 1, 2, ..., n, we consider the star cut sets in order and draw them in the plane. In considering the star cut set at node r, we need only add its incident branches that join r to any of nodes k, where $k > r$. Such branches can, of course, be added if and only if the nodes to be joined have a common mesh or can be made to have a common mesh by the operations discussed previously.

This procedure may sometimes lead to a plane representation more easily than the procedure using circuits. In general, however, the latter procedure appears to be more efficient. For implementing these algorithms by a digital computer, the programmer should consider both procedures.

The theorem can be stated as follows, where we use similar notation as before and designate a star cut set at node r by S_r:

Theorem 2. A graph G is planar if and only if each of the subgraphs G_r, where $r = 1, 2, \ldots, n-2$, is a plane graph and the two nodes of each of the branches of S_{r+1} that must be added lie on a common mesh or can be made to lie on a common mesh by rotation of a subgraph about an articulation node or about the two common nodes of a 2-separator, or by permutation of the subgraphs connected in parallel between the two common nodes of a 2-separator of G_r.

As previously indicated and as will be illustrated in the example, we do not have to add all the branches of a cut set at each step since some have already been added as parts of other preceding cut sets. In addition, it is important to note that we use a slightly different definition of G_r; it is the subgraph obtained after all the branches of S_r have been added and also this subgraph *plus some but not all* of the branches of S_{r+1}.

An Example. We wish to test whether the graph given in Fig. 4 is planar; if it is planar, we desire a plane representation.

We choose the tree shown in Fig. 5a and then add the links in the following order:

$L_1 = (2, 5)$
$L_2 = (6, 8)$
$L_3 = (1, 5)$
$L_4 = (1, 9)$
$L_5 = (4, 9)$
$L_6 = (7, 9)$
$L_7 = (3, 7)$
$L_8 = (1, 4)$
$L_9 = (1, 7)$
$L_{10} = (2, 4)$
$L_{11} = (2, 7)$
$L_{12} = (4, 7)$
$L_{13} = (4, 8)$

We don't go through the formality of listing the meshes for the nodes under consideration because this is evident from the drawing. However, in a computer implementation we would of course list the meshes.

Up through the addition of link L_6 each pair of link nodes has a common mesh so that these links are added without requiring any redrawing of a subgraph; the subgraph G_6 is shown in Fig. 5b. However, nodes 3 and 7 of L_7 do not lie on a common mesh; a possible permutation must therefore be used. All paths between 3 and 7 pass through either node 1 or 4; thus 1 and 4 are the common nodes of a 2-separator. If we rotate the subgraph containing node 3 about nodes 1 and 4, we get the redrawn graph G_6 shown in Fig. 5c, where 3 and 7 lie on the outer mesh. We can now add L_7.

The graph G_7 is now triply connected so that no more permutations or rotations are possible; either each of the node-pairs of the remaining links have common meshes for the subgraph and the augmented subgraphs *as drawn*, or the graph G is nonplanar. We continue to add links L_8 through L_{13}; no crossing of branches occurs, since there exists a common mesh at each step. The final plane graph $G_{F-1} = G$ is shown in Fig. 5d.

We now use this same graph to illustrate how the plane drawing is obtained by use of the star cut-set algorithm. Below we list each of the nodes (except the last) in order, with the adjacent nodes for each node listed after a colon. This of course gives a complete description of the graph. We divide the adjacent nodes into two groups, using a semicolon after those adjacent nodes for which branches will have previously been added. Finally those nodes after the semicolon and the node of the star cut-set itself are underlined if they have already appeared in a preceding cut set and hence have been drawn as part of the existing subgraph; this therefore indicates whether the two nodes of a branch to be added have already been drawn and thus must be examined to see if they lie on a common mesh. It is of course not necessary to include the node labelled n just as it was not necessary to consider mesh F in the first algorithm. We then have in accordance with the above:

1: 2, 4, 5, 7, 9
$\underline{2}$: 1; 3, $\underline{4}$, $\underline{5}$, $\underline{7}$
$\underline{3}$: 2; $\underline{4}$, $\underline{7}$
$\underline{4}$: 1, 2, 3; $\underline{5}$, 6, $\underline{7}$, 8, $\underline{9}$
$\underline{5}$: 1, 2, 4;
$\underline{6}$: 4; $\underline{7}$, $\underline{8}$
$\underline{7}$: 1, 2, 3, 4, 6; $\underline{8}$, $\underline{9}$
$\underline{8}$: 4, 6, 7; $\underline{9}$

From the above we obtain, for example, for S_4 the information that branches to lower-numbered adjacent nodes 1, 2, and 3 have already been drawn, and that branches (4,5), (4,6), (4,7), (4,8) and (4,9) are to be drawn. Of the branches to be drawn we must inspect the two nodes of each of the branches (4,5), (4,7), and (4,9) to see that they are on a common mesh.

We are now ready to attempt to draw the graph. In the subgraphs we encircle the number of a node if all the branches of its star cut-set has already been drawn. Thus if the largest encircled number in a subgraph is r, this indicates that the drawn subgraph is G_r (or G_r plus some but not all branches of S_{r+1} which, as previously indicated, we still designate as G_r), and that we are now considering cut set S_{r+1}.

We draw the branches of S_1, ordering them about node 1 arbitrarily; S_2 is drawn without any required redrawing. The subgraph G_2 is shown in Fig. 6a. However, to add branch (3,4) to S_3, we rotate branch (2,3) about node 2 so that node 3 lies in mesh 25142; subgraph G_2 with (3,4) added is drawn in Fig. 6b. To add branch (3,7) of S_3 we first rotate the subgraph containing node 3 about the 2-separator nodes 1 and 2; G_3 then becomes the graph in Fig. 6c.

To add branch (4,5) of S_4 we rotate the subgraph containing node 5 about nodes 1 and 2. We then can add all the branches of S_4, where (4,6) and (4,8) can be placed arbitrarily. However, a programmer might decide to "look ahead" to obtain information on where to place these branches to minimize redrawings of the graph; he will see, for example, that branches (6,7) and (6,8) appear in S_6 and that (8,9) appears in S_8. He will therefore place (4,6) and(4,8) inside mesh 49174. It is clear, however, that an arbitrary placing of these branches in other meshes would merely require additional rotations or permutations. The subgraph G_4 is shown in Fig. 6d.

Cut set S_5 needs no branches added so that G_5 is the same as G_4 in Fig. 6d except that the number 5 is encircled. Since the nodes of branch (6,7) have a common mesh and so do those of (6,8), cut set S_6 is drawn without requiring any redrawing. Similarly, no redrawing is required for drawing the branches of S_7 and S_8 so that the plane graph $G_8 = G$ is achieved and is shown in Fig. 6e.

The reader will observe that more redrawings were required by use of this method than by the use of a tree and fundamental circuits. However, no serious attempt was made to carry out simplifications that would be thought of by a competent programmer for reducing the computation time.

GRAPHS WITH TERMINAL CONSTRAINTS[†]

For many applications we need to realize a given graph as a plane graph when additional requirements must be met. For example, in a class of microelectronic problems certain nodes of the graph are specified as terminals for making connections to the outside world. No order may be specified for these terminals or a cyclic order may be required. The terminals must lie on the periphery of a semiconductor chip so that connections to other chips or external electrical equipment may be made without using a conductor that goes out of the plane of the chip. If an order is specified, the terminals must lie in the specified order. We characterize this problem as the realization of a plane graph with terminal constraints.

It should first of all be clear that requiring the terminals to lie on the outside of the chip merely means that the specified nodes lie on the outer mesh of the plane graph; they must also be in the prescribed order on this mesh. However, these constraints may make a plane graph no longer realizable without inducing a crossing.

The problem of drawing a graph with terminal constraints is often attacked as if it were a completely new problem. It is fortunate, however, that the solutions given above for drawing a plane graph without terminal constraints can be used in a simple manner to solve this problem; as would be expected, a necessary condition is that the given graph be realizable as a plane graph when no terminal constraints are imposed, that is, the graph without specification of terminals must be planar.

[†] The author is grateful to Mr. John Bruno for useful discussions.

First we consider a triply connected graph and give the necessary and sufficient conditions. This solution, of course, applies to all graphs if we use Mac Lane's procedure for dividing a given graph into a set of triply connected components. However, we also indicate an explicit set of conditions on any connected graph.

The necessary and sufficient conditions for a triply connected graph with terminal constraints to be realizable as a plane graph are:

1) If we assume that no terminal constrains are imposed, the graph is realizable as a plane graph, that is, it satisfies Theroem 1 (or, equivalently Theorem 2).

2) If no order is required on the terminals, the nodes specified as terminals must all lie on one mesh. If an order is specified, the nodes must lie in this cyclic order on the mesh (clockwise or counterclockwise) where nodes on the mesh that are not specified as terminals are disregarded in the ordered set representing the mesh.

The proof should be clear so that we merely sketch it. First since the graph must be planar we conclude that Theorem 1 must be satisfied. Then we make use of the fact that the meshes of a triply connected graph are unique (within a mirror-image reflection). Thus if the terminal nodes do not lie on a mesh, they cannot lie on the outer mesh, since this would contradict the uniqueness of the meshes. If they lie on a mesh but not in the specified order, the same difficulty arises. However, we should note that whether the specified order is clockwise or counterclockwise is immaterial; if a counterclockwise order is required but the realized graph has a clockwise order for the terminals on the mesh, we merely consider the mirror-image of the realized plane graph. If the specified nodes are in the required order and are a proper subset of the nodes on the mesh, we merely consider that the mesh nodes not specified are indented from the periphery of the chip.

The algorithm follows directly. The only part that may need comment is the conversion of an inner mesh to an outer mesh. It is well known that any mesh may serve as the outer mesh in a plane drawing; thus the required graph can be drawn. To see this we merely consider that the plane graph is mapped on a sphere and use stereographic projection, with the north pole lying in the face of the desired outer mesh, to draw the graph in the plane.

As an illustration, we consider the previous example of a triply connected graph shown in Fig. 4. Suppose that nodes 1, 4, 6, and 8 are required as terminals in the clockwise order 1 4 8 6 1. First the graph without terminal constraints is considered, and the plane graph in Fig. 6 is realized. We observe that the specified nodes 1, 4, 6, and 8 do not lie on a single mesh; hence the graph with terminal constraints is not realizable.

If the required terminals and order are specified by 5 4 6 8 9 1 5, that is, the order on a circuit of the plane graph, the required plane graph is again unrealizable.

If the ordered terminals are 1 6 1, the graph is also unrealizable. However, if the terminals are required to have the clockwise order 2 3 4 2, the graph is realizable. This is true because 2, 3, and 4 are on a mesh in a counterclockwise order; to obtain the required clockwise order, we merely use the mirror-image of the graph in Fig. 6e and then draw the graph with the mesh 2 3 4 2 as the outer mesh.

Indeed, a general comment may be made about this graph and similar triangulated graphs. Since every mesh is a triangle, that is, contains three branches, no plane graph is realizable with more than three terminals.

We now briefly consider the connected graph that is not triply connected. The comments made previously about rotation and permutation of subgraphs apply. Again we must first draw a plane graph, if this is possible, for the graph without terminal constraints. If the nodes specified as terminals do not lie on a single mesh in the required order, we then must determine whether they can be made to lie on a common mesh in the required order by rotation of a subgraph about an articulation node or about the common nodes of a 2-separator, or by permutation of subgraphs connected in parallel between the two common nodes of a 2-separator of the plane graph. The procedure for doing this is an extension of that described earlier in connection with the realization of a plane graph without terminal constraints.

In this paper we have been concerned with the topological problem and not electrical equivalences. However, because it is so important for solving this problem when a graph without constraints is realizable as a plane graph, but the graph with constraints is not realizable, we mention that a simple procedure may often be used for satisfying the terminal constraints. This procedure consists of removing any two-terminal subgraph and reconnecting it to the other component of the 2-separator with interchanged common nodes. Thus if such a subgraph is connected to nodes a and b, after disconnecting it we reverse the subgraph and identify its node a (that is, the node that previously coincided with a) with node b, and identify its previous node b with node a of its complement. This gives a graph that is electrically equivalent but only 2-isomorphic to the original graph [3]. It allows us to obtain a new ordering of nodes on a mesh and new meshes; thus without changing the electrical properties of the network corresponding to the graph, we may be able to satisfy constraints that were previously unrealizable.

The general problem of electrical equivalences for satisfying topological constraints requires separate treatment, and so we say no more about it here.

CONCLUSION

Two algorithms have been given for determing whether a graph is planar and for drawing a plane representation if it is planar. Each of the algorithms consists of the repetition of one basic step. Two theorems on which these algorithms are based are stated and proved. These theorems characterize a planar graph, that is, each gives a necessary and sufficient condition for planarity. The algorithms make use of a simple process that is based on the concepts of connectivity of a graph and the fact that a branch cannot be added without a crossover to a graph drawn in the plane if its two nodes do not lie in or on a common mesh. It is believed that the algorithms are simpler than existing algorithms. Computer programs are presently being written based on these algorithms.

These algorithms are then extended to solve the problems of realizing a plane graph with terminal constraints. Here use is made of the property that the meshes of a plane graph are unique within a cyclic inversion (that is, a mirror-image reflection).

Examples are given to illustrate the procedures. Finally, an Appendix is given in which a simple algorithm formulated by Goldstein is presented.

APPENDIX

Here we present an unpublished algorithm due to A.J. Goldstein and illustrate it by two simple examples[†].

At each step of Goldstein's algorithm for drawing a plane graph, there is embedded in the plane a subgraph H of the given graph G. H has the property that it can be extended to a planar embedding of G, if G is planar. H partitions the plane into regions which are numbered and each node of H is labeled with the regions it touches. The nodes and branches of H have the designation "placed." The algorithm is inductive and is initialized by the first four of the six steps given below. Steps 3, 5, and 6 constitute the inductive portion of the algorithm.

(1) First choose an arbitrary circuit H from G and place H in the plane. This partitions the plane into two regions.

(2) If two nodes of H are adjacent in G - H, the complement of H, place the branch joining them in either of the two regions. We now have three regions. If no nodes are adjacent, go to step 4. (If we think of embedding on the sphere, it clearly makes no difference in which region we place the branch, that is, there is no outer or inner region.)

(3) If any other pair of nodes of H are adjacent in G - H, and the nodes have exactly one region in common, then the branch connecting them is forced to lie in that region and it is so placed. Repeat this as often as possible. Finally, all nodes of H that are adjacent in G - H have two or more regions in common. Label branches in G - H connecting such nodes (with two or more regions in common) as "ambiguous". (Of course, if a pair of nodes of H that are adjacent in G - H have no region in common, G is nonplanar.) Now go to step 5.

(4) If no pair of nodes of H are adjacent in G - H, then choose and place any path connecting any pair of nodes of H. The present subgraph H determines three regions. Go to step 3.

(5) Choose a non-placed unambiguous node v. (If there is none, go on to step 6.) Construct any tree T which (a) contains v, (b) is, of course, connected, (c) has placed nodes only as terminal nodes of T, and (d) is maximal. Let B, the boundary of T, be the set of nodes of T which are placed. (i) If B is empty or has only one node, then the embedding of the subgraph spanned by T can be handled as a separate problem. (By the subgraph spanned by T is meant the set of branches both of whose nodes are in T.) Repeat step 5. (ii) If B has more than two nodes and all of them have more than one region in common, then we do not know in which of these regions to place the tree T. Label as ambiguous each non-placed node of T. Repeat step 5. (iii) If the nodes of B have exactly one region R in common, pick any path in T connecting any pair of nodes of B and place it in R. (This placement is forced.) Remove the designation ambiguous from all nodes and branches. Go to step 3. (iv) If the nodes of B have no region in common, then the graph G is nonplanar.

(6) If all non-placed nodes and branches are designated ambiguous, then place any path between any pair of placed nodes. The placement is made in any of the path's permissible regions. Remove the designation ambiguous from all nodes and branches. Now repeat the inductive part of the procedure, namely, steps 3, 5 and 6; that is, go to step 3. If G is planar, the new embedded graph H can be extended to an embedding of G. When all nodes and branches have been placed, the embedding is complete.

[†] The author is grateful to Professor M.A. Harrison, who sent him the abstract of the Goldstein memorandum, from which the algorithm as presented above (with some slight alterations) is taken, and who also sent along some examples.

Two simple examples are given to illustrate the algorithm; one considers a nonplanar graph, whereas the second graph is planar. The first is the nonplanar complete graph K_5, as shown in Fig. 7a. In the first step, we choose the circuit a b c e a, which defines the two regions R_1 and R_2, as shown in Fig. 7b. Then the adjacent nodes a and c are joined, so that regions R_1, R_2, and R_3 are now defined, as shown in Fig. 7c. The nodes b and e are also adjacent; since b touches R_2 and R_3 and e touches R_1 and R_2, (b,e) must lie in the common region R_2.

The subgraph up to this point is shown in Fig. 7d. The nodes and the regions they touch are given below:

a: R_1, R_2, R_3
b: R_2, R_3, R_4
c: R_1, R_3, R_4
e: R_1, R_2, R_4

We note that every pair of nodes has exactly two regions in common. Each branch in the subgraph G - H at this point is ambiguous.

We now choose node d for which the tree T is shown in Fig. 7e. The boundary B for T is the set of terminal nodes, that is,

$$B = \{a, b, c, e\}$$

Since there is no region common to all the nodes of T, the graph is nonplanar by (iv) of step 5 in the algorithm.

The second example, for which we of course assume we are merely given the branches specified by their nodes, is shown in Fig. 8a. We choose the circuit b c d b and observe that nodes b, c, and d all touch region R_1 and R_2, as shown in Fig. 8b. We now choose node a for which the tree is shown in Fig. 8c. The boundary is given by

$$B = \{b, c, d\}$$

As stated in (ii) of step 5, since B has more than two nodes and all of them have more than one region in common, we do not know in which region to place the tree T. We therefore label a as ambiguous and repeat step 5. Since there are no non-placed non-ambiguous nodes, we go to step 6 and place the path b a c in R_2. We now join the adjacent nodes a and d by branch (a, d) to obtain the desired graph.

REFERENCES

1. O. Ore, Theory of Graphs, Providence, R.I.:Amer. Math. Soc. Colloquium Publication, 1962, vol XXXVIII.
2. C. Berge, The Theory of Graphs, New York: John Wiley and Sons, Inc., 1962, (translated by A. Doig).
3. S. Seshu and M.B. Reed, Linear Graphs and Electrical Networks, Reading, Mass.: Addison-Wesley, 1961.
4. A.J. Goldstein, "An Efficient and Constructive Algorithm for Testing Whether a Graph Can Be Embedded in the Plane," Bell Telephone Labs., Murray Hill, New Jersey, 1963, unpublished.
5. G.J. Fisher and O. Wing, "Computer Recognition and Extraction of Planar Graphs from the Incidence Matrix," IEEE Trans. Circuit Theory, vol. CT-13, No. 2, pp. 154-163, June 1966.
6. L. Auslander and L.V. Parter, "On Embedding Graphs in the Sphere," J. Math. Mech., vol. 10, pp. 517-523, May 1961.
7. A. Lempel, S. Even, and I. Cederbaum, "An Algorithm for Planarity Testing of Graphs," presented at the Graph Theory Symposium, Rome, Italy, Aug. 1965.
8. S. Mac Lane, "A Structural Characterization of Planar Combinatorial Graphs," Duke Math. J., vol. 3, No. 3, pp. 460-472, Sept. 1937.

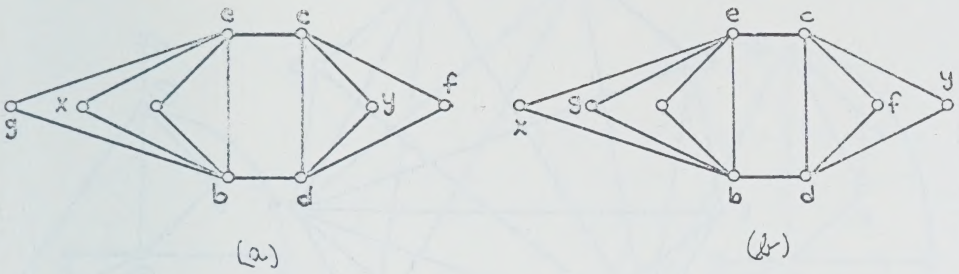

e
c
g
x
y
f
b
d
(a)
e
c
y
x
g
f
b
d
(b)

Fig. 1 - Graphs illustrating permutation of subgraphs to place nodes in a common mesh.

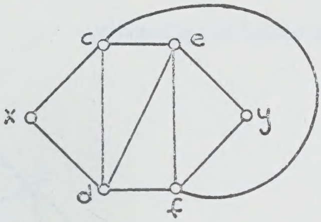


Fig. 2 - Graph which becomes nonplanar if branch (x, y) is added.

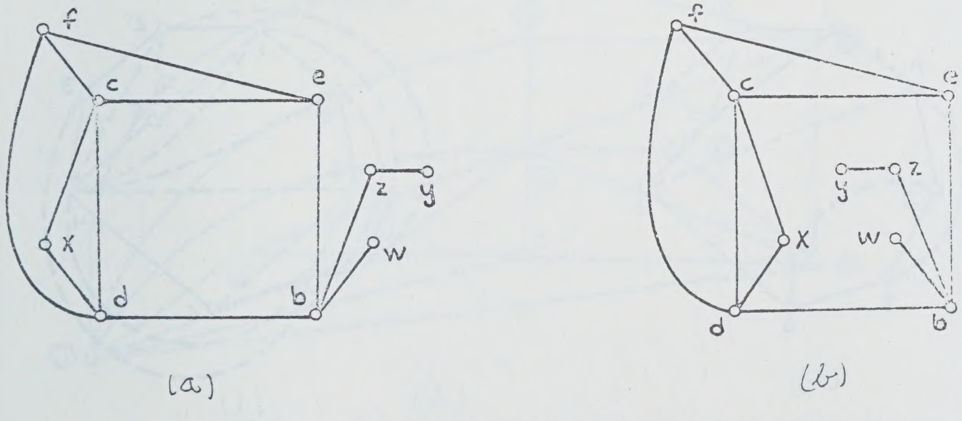

f
c
e
z
y
x
w
d
b
(a)
f
c
e
y
z
x
w
d
b
(b)

Fig. 3 - One-connected graph for which a rotation and permutation yield a common mesh for two nodes to be joined.

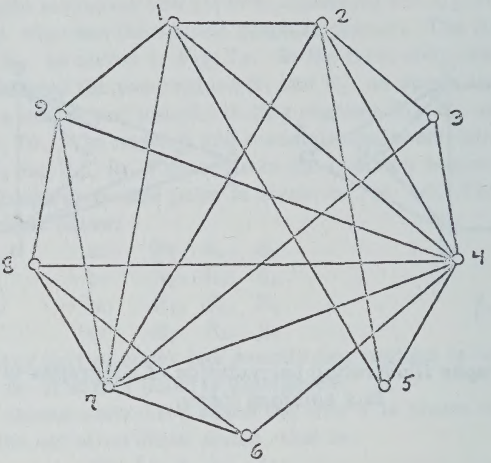


Fig. 4 - Graph to be tested for planarity.

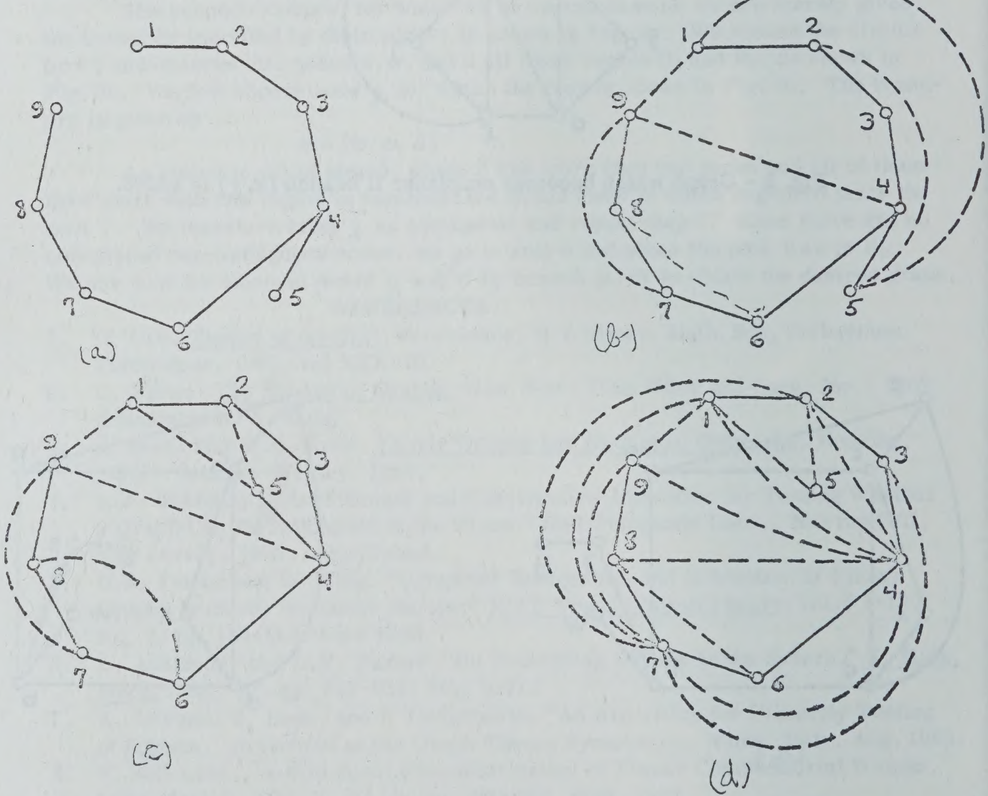

1
2
3
4
5
6
7
8
9
(a)
(b)
(c)
(d)

Fig. 5 - Graphs obtained in drawing the graph of Fig. 4 without crossed branches.

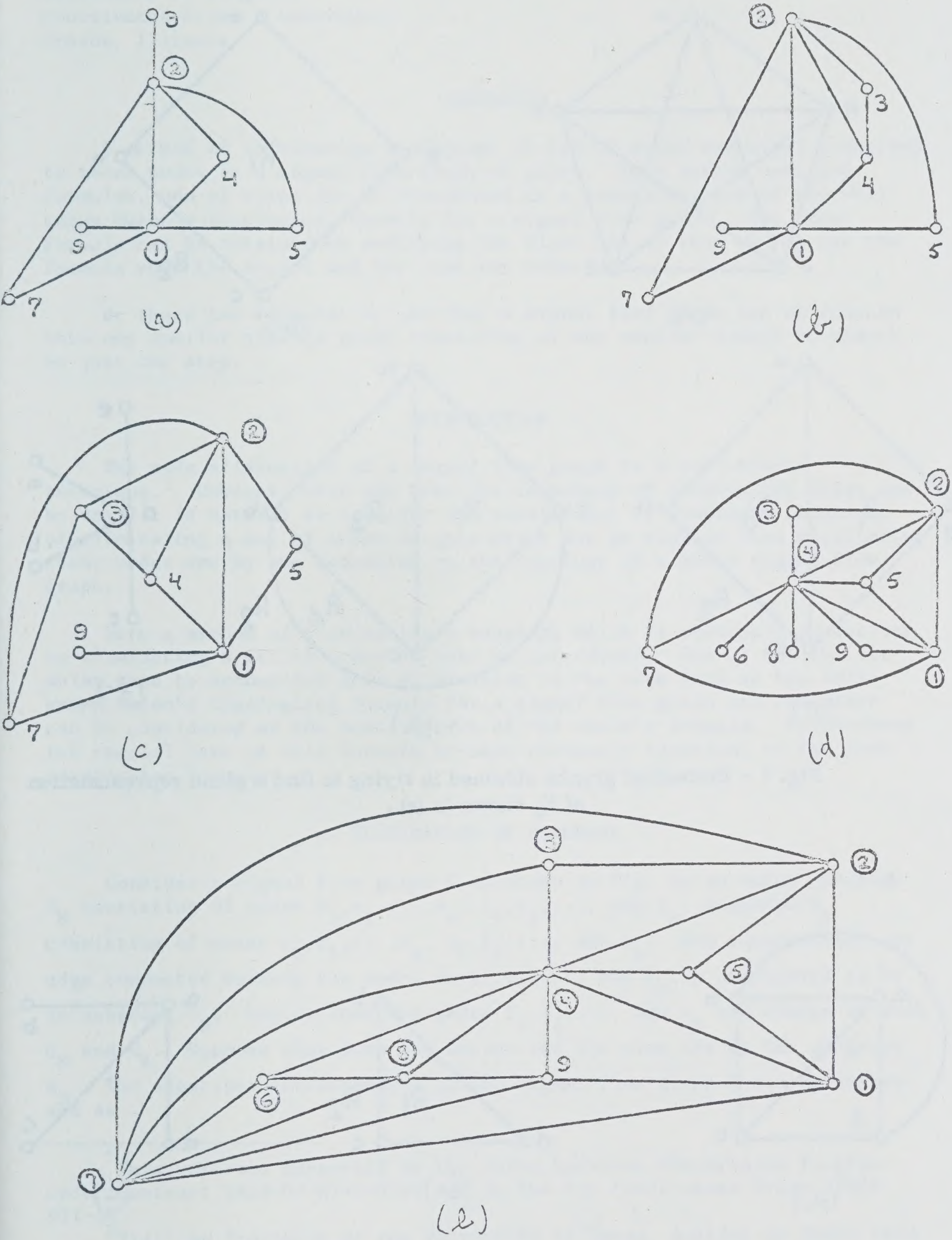

3
②
4
9
①
5
7
(a)
②
3
4
9
①
5
7
(b)
②
③
4
5
9
①
7
(c)
②
③
④
5
6
8
9
7
①
(d)
③
②
⑤
④
⑧
9
⑥
①
⑦
(e)

Fig. 6 - Steps in the plane representation of Fig. 4 by use of the star cut-set algorithm.

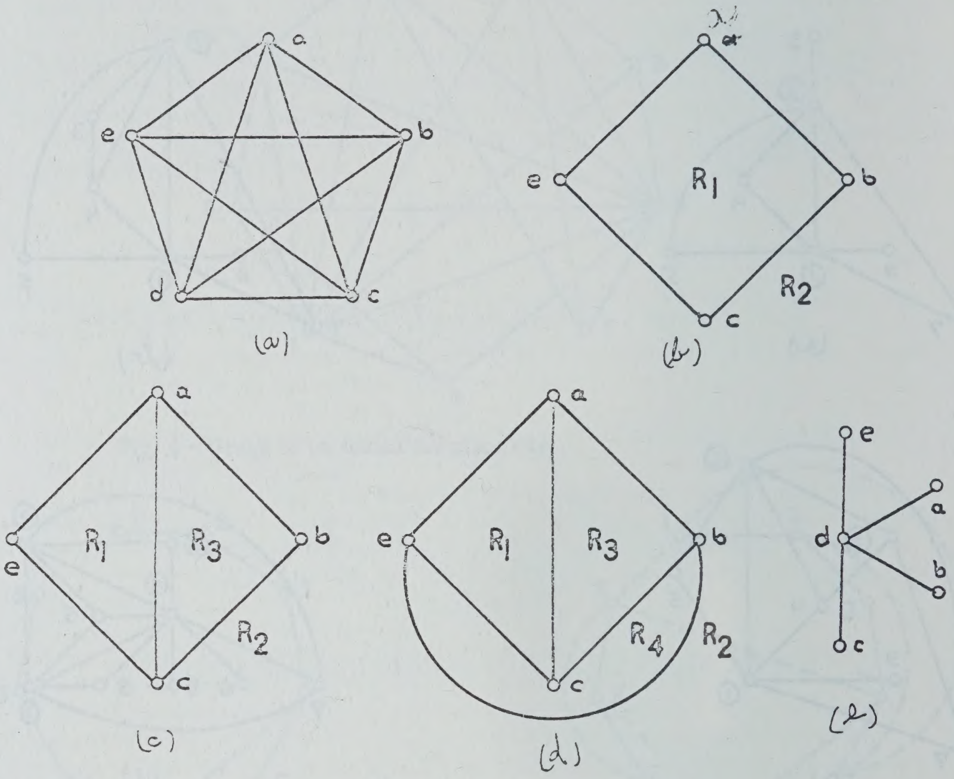

a
b
c
d
e
(a)
a
b
c
e
R1
R2
(b)
a
b
c
e
R1
R3
R2
(c)
a
b
c
e
R1
R3
R4
R2
(d)
e
a
d
b
c
(e)

Fig. 7 - Embedded graphs obtained in trying to find a plane representation of K_5 shown in (a) ;

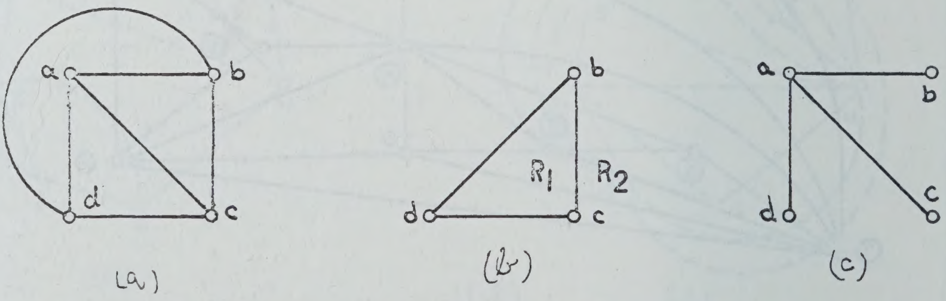

a
b
d
c
(a)
b
d
c
R1
R2
(b)
a
b
d
c
(c)

Fig. 8 - Steps in the realization of a plane graph by use of the algorithm.

ELIMINATION OF SUBGRAPHS IN A SIGNAL FLOW GRAPH[†]

W. MAYEDA[*]
University of Illinois
Coordinated Science Laboratory
Urbana, Illinois

ABSTRACT

A method of eliminating a subgraph (a set of nodes and edges connected to these nodes in a signal flow graph is given. This method uses two formulas, one of which can be considered as a generalization of the well-known Mason's topological formula for a signal flow graph. The other formula can be obtained by modifying the first one so that we can use the formula when the source and the sink are coincided.

By these two formulas we see that a signal flow graph can be reduced into any smaller size (a graph consisting of any smaller number of nodes) by just one step.

INTRODUCTION

The node elimination of a signal flow graph is a well-known technique.[1] However, when one uses the technique of eliminating nodes one by one, it is natural to consider the possibility of finding a technique of eliminating a set of nodes at once which may be simpler than eliminating these nodes one by one depending on the topology of a given signal flow graph.

Here a method of eliminating a subgraph which is obviously equivalent to eliminating a set of nodes at once is introduced. One of the two formulas used to accomplish this elimination is the same form as the well-known Mason's topological formula for a signal flow graph and the other can be considered as the modification of the mason's formula. Furthermore the special case of this formula becomes obviously identical to the node elimination formula.

ELIMINATION OF SUBGRAPH

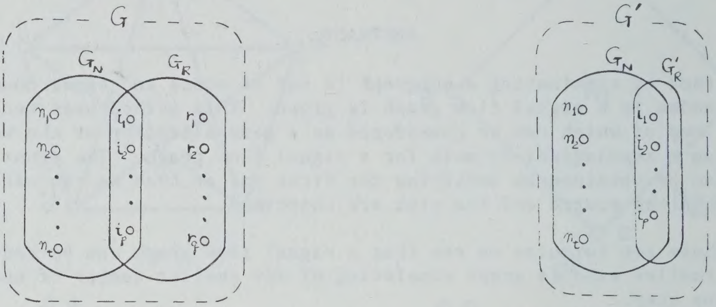
Consider a signal flow graph G as shown in Fig. 1a in which subgraph G_N consisting of nodes $n_1, n_2, \ldots, n_t$, $i_1, i_2, \ldots$, and i_p, subgraph G_R consisting of nodes $r_1, r_2, \ldots, r_q$, $i_1, i_2, \ldots$, and i_p. For convenience, any edge connected between the nodes in $i_1, i_2, \ldots$, and i_p is considered to be in subgraph G_N. Notice that the nodes $i_1, i_2, \ldots$, and i_p are common in both G_N and G_R. Suppose that both the source and the sink are in the subgraph G_N. The equations corresponding to the signal flow graph G can be expressed as

---

[†] This work was supported by the Joint Services Electronics Program under Contract DAAB-07-67-C-0199 and by the Air Force under Grant AFOSR 931-66.

[*] Visiting Professor at the University of Texas, Austin; on leave from the University of Illinois, Urbana.

$$\begin{bmatrix} A_{11} & A_{12} & 0 \\ A_{21} & A_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_N \\ X_I \\ X_R \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & B_{11} & B_{12} \\ 0 & B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} X_N \\ X_I \\ X_R \end{bmatrix} \begin{bmatrix} \underset{\sim}{X}_N \\ X_I \\ X_R \end{bmatrix} \tag{1}$$



(a) A Signal Flow Graph

(b) After Eliminating Sub-Graph G_R

Figure 1

where the matrix

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{2}$$

is from subgraph G_N and the matrix B

$$B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \tag{3}$$

is from subgraph G_R. The vectors X_N, $\underset{\sim}{X}_N$, X_I, and X_R are respectively

$$X_N = \begin{bmatrix} X_{n_1} \\ X_{n_2} \\ \vdots \\ X_{n_t} \end{bmatrix}, \quad \underset{\sim}{X}_N = \begin{bmatrix} X'_{n_1} \\ X_{n_2} \\ \vdots \\ X_{n_t} \end{bmatrix}, \quad X_I = \begin{bmatrix} X_{i_1} \\ \vdots \\ X_{i_p} \end{bmatrix}, \quad X_R = \begin{bmatrix} X_{r_1} \\ \vdots \\ X_{r_q} \end{bmatrix}, \tag{4}$$

where X_k is the variable (weight) given to node k for $k = n_1, \ldots, n_t, i_1, \ldots, i_p, r_1, \ldots, r_q$, X_{n_1} is the source and X'_{n_1} is the sink.

Equation 1 can be written as the following two equations:

$$\begin{bmatrix} A_{11} & A_{12} \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} = \underset{\sim}{X}_N \tag{5}$$

$$\begin{bmatrix} A_{21} & A_{22} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} X_I \\ X_R \end{bmatrix} = \begin{bmatrix} X_I \\ X_R \end{bmatrix} . \qquad (6)$$

Furthermore Eq. (6) can be modified as

$$\begin{bmatrix} A_{21} & A_{22} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} + \begin{bmatrix} B_{11}-U & B_{12} \\ B_{21} & B_{22}-U \end{bmatrix} \begin{bmatrix} X_I \\ X_R \end{bmatrix} = 0. \qquad (7)$$

For convenience, let

$$\underline{B} = \begin{bmatrix} B_{11}-U & B_{12} \\ B_{21} & B_{22}-U \end{bmatrix} . \qquad (8)$$

Suppose $\underline{B}$ is nonsingular, then from Eq. (7), we have

$$\underline{B}^{-1} \begin{bmatrix} A_{21} & A_{22} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} + \begin{bmatrix} X_I \\ X_R \end{bmatrix} = 0. \qquad (9)$$

Let

$$\underline{B}^{-1} = \begin{bmatrix} (\underline{B}^{-1})_{11} & (\underline{B}^{-1})_{12} \\ (\underline{B}^{-1})_{21} & (\underline{B}^{-1})_{22} \end{bmatrix} .$$

Then Eq. (9) can be written in two equations

$$(\underline{B}^{-1})_{11} \left[A_{21} \quad A_{22}\right] \begin{bmatrix} X_N \\ X_I \end{bmatrix} + X_I = 0 \qquad (10)$$

$$(\underline{B}^{-1})_{21} \left[A_{21} \quad A_{22}\right] \begin{bmatrix} X_N \\ X_I \end{bmatrix} + X_R = 0. \qquad (11)$$

If $(\underline{B}^{-1})_{11}$ is nonsingular, we can write Eq. (10) as

$$\left[A_{21} \quad A_{22}\right] \begin{bmatrix} X_N \\ X_I \end{bmatrix} + {(\underline{B}^{-1})_{11}}^{-1} X_I = 0 \qquad (12)$$

or

$$\left[A_{21} \quad A_{22}\right] \begin{bmatrix} X_N \\ X_I \end{bmatrix} + \left[{(\underline{B}^{-1})_{11}}^{-1} + U\right] X_I = X_I. \qquad (13)$$

This with Eq. (5) gives

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & (\underline{B}^{-1})_{11}{}^{-1} + U \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} = \begin{bmatrix} \underset{\sim}{X}_N \\ X_I \end{bmatrix} \tag{14}$$

from which the solution can be obtained. Thus, the signal flow graph after eliminating subgraph G_R must be the one which is represented by Eq. (14). This equation shows that we can replace subgraph G_R by a new signal flow graph which consists of nodes $i_1, i_2, \ldots,$ and i_p and is given by the equation

$$[(\underline{B}^{-1})_{11}{}^{-1} + U]\, X_I = X_I. \tag{15}$$

Thus we only need to discuss each entry of $[(\underline{B}^{-1})_{11}{}^{-1} + U]$ so that we can obtain the new subgraph from the given subgraph G_R.

Entry of $[(\underline{B}^{-1})_{11}{}^{-1} + U]$

For a given matrix B, let the symbol $B\begin{pmatrix} t_1 t_2 \ldots t_v \\ u_1 u_2 \ldots u_v \end{pmatrix}$ or $B\begin{pmatrix} t_m \\ u_m : m=1,2,\ldots v \end{pmatrix}$ be the submatrix obtained from B by deleting rows $t_1, t_2, \ldots,$ and t_v and columns $u_1, u_2, \ldots,$ and u_v. Further, we use the symbol $C_{k\ell}$ for each entry of $(\underline{B}^{-1})_{11}{}^{-1}$, i.e.,

$$(\underline{B}^{-1})_{11}{}^{-1} = [C_{k\ell}]. \tag{16}$$

Also from Eqs. (1) and (3), the rows and columns of B corresponding to x_{i_m} (and $x_{r_{m'}}$) be i_m and $r_{m'}$. Then*

$$C_{i_t i_u} = \frac{\left|[B-U]\begin{pmatrix} i_m : & m=1,2,\ldots p, m\neq t \\ i_m & m'=1,2,\ldots p, m'\neq u \end{pmatrix}\right|}{\left|[B-U]\begin{pmatrix} i_w : & w=1,2,\ldots,p \\ i_w & \end{pmatrix}\right|}. \tag{17}$$

---

*Let $M = [m_{ij}]$ and Δ_{ij} be the cofactor at i-j position of M.

Then $$\begin{vmatrix} \Delta_{i_1 j_1} & \ldots & \Delta_{i_1 j_r} \\ \ldots & & \\ \Delta_{i_1 j_1} & \ldots & \Delta_{i_r j_r} \end{vmatrix} = |M|^{r-1} \left| M\begin{pmatrix} i_1 \ldots i_r \\ j_1 \ldots j_r \end{pmatrix} \right|$$

where $M\begin{pmatrix} i_1 \ldots i_r \\ j_1 \ldots j_r \end{pmatrix}$ is obtained from M by deleting rows $i_1 \ldots i_r$ and columns $j_1 \ldots j_r$.

The graph corresponding to $B_{\substack{i_w: w=1,2,\dots,p \\ i_w}}$ is $G_{R(i_w:w=1,2,\dots,p)}$ which is obtained from G_R by deleting nodes $i_1, i_2, \dots,$ and i_p and all edges connected to these nodes because the deletion of row i_w and column i_w is equivalent to deleting node i_w and all edges connected to the node in G_R. Thus[2]

$$\left|[B-U]\begin{pmatrix} i_w: w=1,2,\dots,p) \\ i_w \end{pmatrix}\right| = (-1)^q \sum_{(j)} (-1)^h \text{ (P-set cycle product j) of } \bar{G}_{R(i_w:w=1,2,\dots,p)} \qquad (18)$$

where $\bar{G}_{R(i_w:w=1,2,\dots,p)}$ is a Coate's flow graph obtained from signal flow graph $G_{R(i_w:w=1,2,\dots,p)}$ by inserting one self loop with -1 as its weight to every node in $G_{R(i_w:w=1,2,\dots,p)}$ and h is the number of direct circuits in P-sec j. Thus[3]

$$\left|[B-U]\begin{pmatrix} i_w: w=1,2,\dots,p \\ i_w \end{pmatrix}\right| = (-1)^q \{1-\Sigma P_{m_1} + \Sigma P_{m_2} \dots\} \text{ of } G_{R(i_w:w=1,2,\dots,p)} \qquad (19)$$

where P_{m_1} is the product of the weight of all edges in the m^{th} set of i non-touching directed circuits.

Similarly if $i_r \neq i_u$

$$\left|[B-U]\begin{pmatrix} i_m: m=1,2,\dots,p, m\neq t \\ i_{m'}, m'=1,2,\dots,p, m'\neq u \end{pmatrix}\right| = (-1)^q \sum_k D_{(i_t i_u)k}\Delta_k \text{ of } G_{R(i_{w'}:w'=1,2,\dots,p,w'\neq t,u)} \qquad (20)$$

where $D_{(i_t i_u)}$ is the product of weights of edges in directed path k from node i_t to node i_u and Δ_k is $(1-\Sigma P_{m_1} + \Sigma P_{m_2} \dots)$ of the subgraph obtained from $G_{R(i_w:w'=1,2,\dots,p,w'\neq t,u)}$ by deleting all nodes in the directed path k and all edges connected to the nodes. Hence directed path k and any directed circuits in P_{m_1} in Δ_k must not have any common nodes. Hence Eq. (17) becomes

$$C_{i_t i_u} = \frac{\Sigma D_{(i_t i_u)k}\Delta_k \text{ of } G_{R(i_{w'}:w'=1,2,\dots,p,w'\neq t,u)}}{1-\Sigma P_{m_1} + \Sigma P_{m_2} \dots \text{ of } G_{R(i_w:w=1,2,\dots,p)}} . \qquad (21)$$

When i_t and i_u are considered to be the source and the sink of $G_{R(i_{w'}:w'=1,2,\dots,p,w'\neq t,u)}$ the formula in Eq. (21) is the Mason's topological formula. When $i_t = i_u$, $\left|[B-U]_{\begin{pmatrix} i_m : m=1,2,\ldots,p,m\neq t \\ i_{m'} : m'=1,2,\ldots,p,m'\neq u \end{pmatrix}}\right|$ becomes $\left|[B-U]_{(i_m : m=1,2,\ldots,p,m\neq t)}\right|$. Thus the formula becomes the same as that given by Eq. (18). However, because of a self-loop of -1 at node i_t, we can write

$$\begin{aligned} \left|[B-U]_{(i_m:m=1,2,\ldots,p:m\neq t)}\right| &= (-1)^{q+1}\Sigma(-1)^h \text{P-set cycle product } j \\ &\quad \text{of } \bar{G}_{R(i_w:w=1,2,\ldots,p,w\neq t)} \\ &= (-1)^{q+1}\{F_1 + F_2\} \end{aligned} \tag{22}$$

where F_1 is the collection of all terms contained in the self loop of -1 at node i_t and F_2 is the remaining terms.

Since every term in F_1 contains this particular self loop, F_1 can be expressed as

$$F_1 = (-1)\{\Sigma(-1)^h \text{P-set cycle product } j \text{ of } \bar{G}_{R(i_w:w=1,2,\ldots,p)}\}. \tag{23}$$

Thus Eq. (17) with $i_t = i_u$ becomes

$$\begin{aligned} C_{i_t i_t} &= \frac{(-1)\{1-\Sigma P_{m_1} + \Sigma P_{m_2} \ldots \text{ of } G_{R(i_w:w=1,2,\ldots,p)}\} + \sum_k C(i_t)_k \Delta_{k'}}{(1-\Sigma P_{m_1} + \ldots) \text{ of } G_{R(i_w:w=1,2,\ldots,p)}} \\ &\qquad\qquad \text{of } G_{R(i_{w'}:w'=1,2,\ldots,p,w'\neq t)} \end{aligned} \tag{24}$$

where $C_{(i_t)k}$ is the product of weight of all edges in directed circuit k, which contains node i_t. Also Δ'_k is $(1-\Sigma P_{m_1} + \Sigma P_{m_2} \ldots)$ of the subgraph obtained from $G_{R(i_w:w'=1,2,\ldots,p,w'\neq t)}$ by deleting all nodes in the directed circuit k and all edges connected to these deleted nodes.

Equation (24) can be written as

$$C_{i_t i_t} = \frac{-1 + \sum_k C_{(i_t)k}\Delta'_k \text{ of } (G_{R(i_{w'}:w'=1,2,\ldots,p,w'\neq t)})}{1 - \Sigma P_{m_1} + \Sigma P_{m_2} \ldots \text{ of } G_{R(i_w:w=1,2,\ldots,p)}}. \tag{25}$$

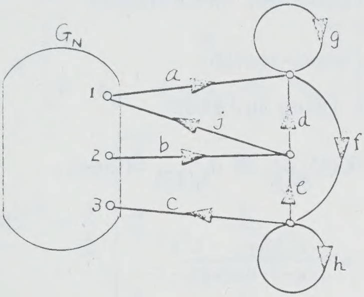
Hence the (i_t, i_t) entry of $k_{i_t i_t}$ of $[(\underline{B}^{-1})_{11}^{-1} + U]$ (because of U) becomes

$$k_{i_t i_t} = \frac{\Sigma\, C_{(i_t)k}\Delta'_k \text{ of } (G_{R(i_{w'}:w'=1,2,\ldots,p,w'\neq t)})}{1 - \Sigma P_{m_1} + \Sigma P_{m_2} \ldots \text{ of } G_{R(i_w:w=1,2,\ldots,p)}} \tag{26}$$

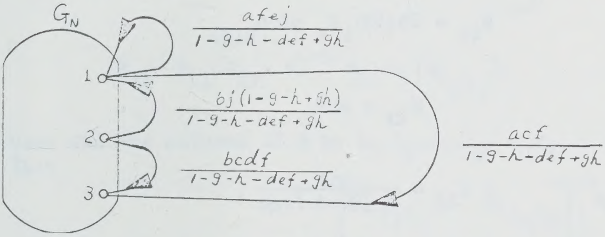
and from Eq. (21), the (i_t, i_u) entry $k_{i_t i_u}$ of $[(\underline{B}^{-1})_{11}^{-1} + U]$ becomes

$$k_{i_t i_u} = C_{i_t i_u} = \frac{\Sigma D_{(i_t i_u)k} \Delta_k \text{ of } G_{R(i_{w'}:w'=1,2,\ldots,p,w'\neq t,u)}}{1 - \Sigma P_{m_1} + \Sigma P_{m_2} \ldots \text{ of } G_{R(i_w:w=1,2,\ldots,p)}} \quad . \tag{27}$$

Equations (26) and (27) give two formulas by which a set of nodes can be eliminated at once from a signal flow graph.

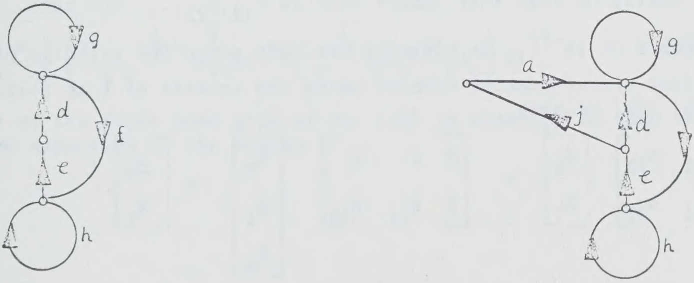

G_N
1
2
3
a
j
b
c
g
d
f
e
h

(a) A Given Signal Flow Graph


G_N
$\frac{afej}{1-g-h-def+gh}$
1
$\frac{bj(1-g-h+gh)}{1-g-h-def+gh}$
2
$\frac{bcdf}{1-g-h-def+gh}$
3
$\frac{acf}{1-g-h-def+gh}$

(b) Resultant Signal Flow Graph After Eliminating G_R

Figure 2. Signal Flow Graph


g
d
f
e
h
a
g
j
d
f
e
h

(a) $G_{R(123)}$ (b) $G_{R(23)}$

Figure 3

Example: Suppose we have a signal flow graph shown in Figure 2a in which the source and the sink are in G_N. Then we can eliminate subgraph G_R by the following steps:

Step 1: Since the denominator D of both Eq. (26) and (27) are the same, which is given by

$$D = 1 - \Sigma P_{m_1} + \Sigma P_{m_2} \ldots \text{ of } G_{R(1,2,3)}$$

where $G_{R(1,2,3)}$ is given in Figure 3a. D is simply

$$D = 1-g-h-def+gh.$$

For k_{11}, the numerator N_{11} is (from Eq. (26))

$$N_{11} = \Sigma C(1)_k \Delta'_k \text{ of } G_{R(23)} = aefj.$$

Thus

$$k_{11} = \frac{aefj}{1-g-h-def+gL} .$$

For k_{13}, numerator N_{13} is (from Eq. (27))

$$N_{13} = \Sigma D(12)_k \Delta_k \text{ of } G_{R(2)}$$

or

$$N_{13} = acf.$$

Thus

$$k_{13} = \frac{acf}{1-g-h-def+gh} .$$

Similarly

$$k_{23} = \frac{bcdf}{1-g-h-def+gh}$$

and

$$k_{21} = \frac{bj(1-g-h+gh)}{1-g-h+def+gh} .$$

It can easily be seen that others such as $k_{12}, k_{22}, \ldots$ are zero.

When B or $(B^{-1})_{11}$ is singular, we must prove the validity of the equivalent transformation without using the inverse of B as follows: From Eq. (1), we have

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ B & B_{11} & B_{12} \end{bmatrix} \begin{bmatrix} X_N \\ X_I \\ X_R \end{bmatrix} = \begin{bmatrix} \underset{\sim}{X}_N \\ X_I \end{bmatrix} \tag{28}$$

and

$$[B_{21} \quad B_{22}] \begin{bmatrix} X_I \\ X_R \end{bmatrix} = X_R \, . \tag{29}$$

Solving Eq. (29) under the assumption that B_{22}-U is non-singular, we have

$$X_R = -[B_{22} - U]^{-1} \, B_{21} \, X_I . \tag{30}$$

The substitution of this into Eq. (28) gives

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & B_{11} - B_{12}[B_{22}-U]^{-1}B_{21} \end{bmatrix} \begin{bmatrix} X_N \\ X_I \end{bmatrix} = \begin{bmatrix} \underset{\sim}{X}_N \\ X_I \end{bmatrix} . \tag{31}$$

Let

$$[B_{22}-U]^{-1} = \frac{1}{\Delta} \begin{bmatrix} \Delta_{r_1 r_1} & \Delta_{r_2 r_1} & \cdots & \Delta_{r_q r_1} \\ \Delta_{r_1 r_2} & \Delta_{r_2 r_2} & \cdots & \Delta_{r_q r_2} \\ \cdots & \cdots & \cdots & \cdots \\ \Delta_{r_1 r_q} & \Delta_{r_2 r_q} & \cdots & \Delta_{r_q r_q} \end{bmatrix} \tag{32}$$

and

$$B_{11} - B_{12}[B_{22} - U]^{-1} \, B_{21} = [m_{i_t i_u}] . \tag{33}$$

Also let the rows and the columns of B be $i_1, i_2, \ldots i_p$ $r_1 r_2 \ldots$ and r_q in that order. Then

$$m_{i_t i_u} = b_{i_t i_u} - \frac{1}{\Delta} \left[b_{i_t r_1} \, b_{i_t r_2} \ldots b_{i_t r_q} \right] \begin{bmatrix} \Delta_{r_1 r_1} & \cdots & \Delta_{r_q r_1} \\ \Delta_{r_1 r_2} & \cdots & \Delta_{r_q r_2} \\ & \cdots & \\ \Delta_{r_1 r_q} & \cdots & \Delta_{r_q r_q} \end{bmatrix} \begin{bmatrix} b_{r_1 i_u} \\ b_{r_2 i_u} \\ \vdots \\ b_{r_q i_u} \end{bmatrix} \tag{34}$$

or

$$m_{i_t i_u} = \frac{1}{\Delta} \left(b_{i_t i_u} \Delta - \sum_{\omega, \zeta = 1}^{q} b_{i_t r_\omega} \Delta_{r_\zeta r_\omega} \, b_{r_\zeta i_u} \right) . \tag{35}$$

The numerator of the right hand side of Eq. (35) is exactly the same as the row-column expansion of the matrix

$$\left[\begin{array}{c|c} b_{i_t i_u} & b_{i_t r_1} \ldots b_{i_t r_q} \\ \hline b_{r_1 i_u} & \\ b_{r_2 i_u} & B_{22} - U \\ \vdots & \\ b_{r_q i_u} & \end{array}\right] = [B-U]\begin{pmatrix} i_m : m=1,2,\ldots,p, m \neq t \\ i_{m'} : m'=1,2,\ldots,p, m' \neq u \end{pmatrix} . \tag{36}$$

Thus from Eq. (17), and Eq. (27)

$$m_{i_t i_u} = C_{i_t i_u} = k_{i_t i_u} . \tag{37}$$

When t = u, Eq. (35) becomes

$$m_{i_t i_t} = \frac{1}{\Delta}\left(b_{i_t i_t} \Delta - \sum_{\omega\zeta=1}^{q} b_{i_t r_\omega} \Delta_{r_\zeta r_\omega} b_{r_\zeta i_t} \right) . \tag{38}$$

Thus

$$m_{i_t i_t} = \frac{\left|\begin{array}{c|c} b_{i_t i_t} & b_{i_t r_1} \quad b_{i_t r_2} \quad \ldots \quad b_{i_t r_q} \\ \hline b_{r_1 i_t} & \\ b_{r_2 i_t} & B_{22} - U \\ \vdots & \\ b_{r_q i_t} & \end{array}\right|}{|B_{22} - U|} . \tag{39}$$

By modifying this as

$$m_{i_t i_t} = m_{i_t i_t} - 1 + 1$$

$$\frac{\left|\begin{array}{c|c} b_{i_t i_t} & b_{i_t r_1} \ldots b_{i_t r_q} \\ \hline b_{r_1 i_t} & \\ \vdots & B_{22} - U \\ b_{r_q i_t} & \end{array}\right| - |B_{22} - U|}{|B_{22} - U|} + 1$$

$$
= \frac{\left|\begin{array}{c|ccc} b_{i_t i_t} - 1 & b_{i_t r_1} & \cdots & b_{i_t r_q} \\ \hline b_{r_1 i_t} & & & \\ b_{r_2 i_t} & & & \\ \vdots & & B_{22} - U & \\ b_{r_q i_t} & & & \end{array}\right|}{|B_{22} - U|} + 1
$$

$$
\frac{\left|[B-U]\Big/\begin{pmatrix} i_m : m=1,2,\ldots,p, m \neq t \\ i_m \end{pmatrix}\right|}{\left|[B-U]\Big/\begin{pmatrix} i_m : m=1,2,\ldots,p \\ i_m \end{pmatrix}\right|} + 1 \quad . \qquad (40)
$$

Thus from Eqs. (22), (25), and (26),

$$
m_{i_t i_t} = k_{i_t i_t} \quad . \qquad (41)
$$

This proves that the equivalent transformation is valid if B_{22}-U is non-singular. Since B_{22}-U must be non-singular to have at least one of $k_{i_t i_u}$ exists, this assumption is valid also. Notice that we assume at that $\underset{\sim}{X}_N$ contains sink $X_{n_1'}$, and X_N contains source X_{n_1}. However, the proof would be the same when the source or (and) the sink are in X_1.

CONCLUSION

We show that the simple formulas given by Eqs. (26) and (27) are enough to eliminate a set of nodes in a signal flow graph. It can be seen that for many cases, the technique will be simpler than the elimination of nodes one by one. Notice that the assumption about the non-singularities of submatrices are valid as long as all edges which replace subgraph G_R have finite weights.

REFERENCES

1. S. J. Mason, "Feedback Theory - Some Properties of Signal Flow Graphs," Proc. of IRE, 41, 1144-1156, September 1953.

2. C. L. Coates, "Flow Graph Solutions of Linear Algebraic Equations," IRE Trans. on Circuit Theory, CT-6, 170-187, June 1959.

3. S. J. Mason, "Topological Analysis of Linear Non-Reciprocal Network," Proc. of IRE, 45, 829-838, June 1957.

ON THE RELATIONSHIP BETWEEN FLOW-GRAPHS AND THE TRAVELING SALESMAN PROBLEM

E. S. DAVIDSON and G. METZE
Coordinated Science Laboratory
University of Illinois
Urbana, Illinois

ABSTRACT

It is the intention of this paper to exploit a correspondence between connections of a flow-graph and solutions of a traveling salesman problem. A tighter bound is derived for the branch-and-bound algorithm for the traveling salesman problem as well as an upper bound on the number of solutions containing a given partial solution. On the other hand, a constructive and economical tree representation for the connections of a flow-graph is found in which the connections and desired one-connections may be displayed in a single tree. Further, if approximate solutions are sufficient, large gain connections may be generated first. A bound on the error due to neglected connections is available.

FLOW-GRAPH DEFINITION

The flow-graphs given here are of the Coates^2 type; the definition essentially follows that of Desoer^4. A flow-graph, G, may be used to represent and solve an ordered set of simultaneous linear equations of the form:

$$\sum_{j=1}^{n} a_{kj} x_j - b_k = 0 \qquad (k = 1,2,\ldots,n),$$

which arises for example in network theory. A node is present in G for each variable, x_j; there is a "source node", C, for the constants (b's). For each non-zero a_{kj} a branch is present directed to node x_k from node x_j and assigned weight (branch gain) a_{kj}. For each non-zero b_k a branch is present directed to node x_k from the constant node C and assigned weight $-b_k$ (cf. matrix and flow-graph of Fig. 2). It should be mentioned that the ordering of the equations, reflected in subscript k, influences the form of the flow-graph in a non-trivial manner.

Let G_o be a subgraph of G formed by deleting node C. A connection is a subgraph of G_o having exactly one branch originating from and one branch terminating at each node. A connection is thus seen to be a set of disjoint cycles or loops involving all variable nodes. The sign of connection ρ is the sign of $(-1)^{n+L_\rho}$ where n is the number of variable nodes and L_ρ the number of loops in connection ρ. The gain of connection ρ, C_ρ, is the product of the branch gains of each branch in the connection. The denominator determinant, Δ, is the algebraic sum of the connection gains over all connections (ρ) of the derived flow-graph G_o.

$$\Delta = (-1)^n \sum_\rho (-1)^{L_\rho} C_\rho \ (G_\rho)$$

A one-connection from C to j is a subgraph of G having one branch originating from C, one branch terminating at x_j, and exactly one branch originating from and one terminating at each other node of G. A one-connection from C to j is thus seen to be a forward path from C to j and a disjoint set of cycles involving all nodes not in the path. The sign of one-connection σ is the sign of $(-1)^{n+L_\sigma}$ where L_σ is the number of complete loops in one-connection σ. The gain of one-connection σ, C_σ, is the product of the branch gains of each branch in the one-connection. The numerator determinant, Δ^j, for x_j is the algebraic sum of the gains of one-connections from C to j of the flow graph G:

$$\Delta^j = (-1)^n \sum_\sigma (-1)^{L_\sigma} C_\sigma \ (G{:}C \rightarrow j)$$

By Cramer's rule for solving simultaneous linear equations: $x_j = \Delta^j/\Delta$.

TRAVELING SALESMAN PROBLEM DEFINITION

The traveling salesman problem, classically stated, involves a salesman who must leave home, visit each of a number of cities exactly once, and return home. In what order should he visit the cities to minimize the total distance traveled? This problem has challenged mathematicians because its statement is easy and its solution is difficult, and it has challenged engineers because it arises in path minimization[3] and scheduling. The problem may be represented by a distance matrix, D, where d_{ij} is the distance from city i to city j. Alternatively, it may be represented by a graph in which each node corresponds to a city and a branch is present for each possible city-to-city trip, weighted with the appropriate distance. It should be noted that the distances need not be symmetric ($d_{ij} \neq d_{ji}$) and that travel between some pairs of cities, in particular self-loops, may be forbidden ($d_{ij} = \infty$).

A tour is then a single loop connection of the graph, the length of the tour is the sum of the distances of the branches in the tour, and the shortest, i.e. minimum length, tour is to be found. The "length" of a connection thus appears to be different from the "gain" of a connection, computed as the product of branch gains. The two may, however, be made equivalent provided branch gains are taken as antilogs of distances. Then the optimum solution of a traveling salesman problem is the single loop connection having minimum gain.

Using this correspondence, one can now consider applying methods developed for flow-graphs to traveling salesman graphs, and vice versa. Of particular relevance is the branch-and-bound algorithm for solution of the traveling salesman problem[6]. A few preliminary concepts must be described for this algorithm. Consider the distance matrix of a traveling salesman problem. If a number, θ, is subtracted from each element of a row or column, the length of every tour is reduced by θ (since every tour must contain exactly one element from each row and column). The ordering of the tours by length therefore remains unchanged. The minimum element of each row and column may be made 0 by subtracting appropriate θ's from rows *and* columns of the distance matrix. The resulting matrix is called a *reduced matrix*. The sum of these θ's forms a *lower bound* on the length of all tours of the original problem, since any tour of the reduced matrix is non-negative. A number, Δ, is defined for each row and column as the second smallest entry in that row or column.

The algorithm may now be defined with the help of Fig. 1. The original distance matrix is given in Fig. 1a, and the reduced matrix in Fig. 1b. The sum of the θ's, 48, forms the lower bound on the length of all tours. A solution tree is constructed, Fig. 1g, with the node ALL representing all tours. This node has a *bound* of 48. A 0 element is then selected for *branching* from the ALL node. The sum of the row and column Δ's of the selected element is to be a maximum. In this case element 1-4 with a Δ sum of 8 is selected. Two nodes are added to the solution tree: 1-4 and $\overline{1-4}$, Fig. 1g, representing all tours containing 1-4, and all tours not containing 1-4, respectively. For the purpose of branching from the 1-4 node, the matrix is modified by removing the 1 row and 4 column, by setting the prohibited loop-forming element 4-1 to ∞, and by reducing the resulting matrix. The bound is then increased by 1 (the sum of the new θ's) to 49, Fig. 1c. For the purpose of branching from the $\overline{1-4}$ node, the element 1-4 is set to ∞ and the matrix reduced, thereby increasing the bound by 8 (the sum of the Δ's of 1-4) to 56, Figs. 1f and 1g. Thus, the element i-j, whose $\overline{i-j}$ node has a maximum bound, is selected. This process of element selection, tree branching, matrix modification, and bound increasing is repeated until a node of the tree represents a single tour with length Z. All nodes with bound $\geq$ Z cannot produce a better tour than the one found; but those with bound $<$ Z must be investigated further. Thus, in the example, Fig. 1g, after four branching steps are undertaken, the upper part of the tree is developed and a solution of length 61 is found.

Each other terminal node of the tree has a bound higher than 61 except for the node labeled $\overline{1-4}$ with bound 56. If the algorithm is continued, five additional branching steps are required to develop the remainder of the tree. It is only then discovered that every tour lacking branch 1-4 has length at least 62. Thus, the first tour found is the shortest tour. Note that in this example, more time was spent establishing that a tour already found was indeed optimal than in finding that optimal tour. This fact suggests that a tighter bound might have been applied at some points in the solution process to reduce the number of necessary branchings significantly. Such a bound is in fact suggested by analogy to flow-graphs.

APPLICATIONS OF FLOW-GRAPHS TO THE TRAVELING SALESMAN PROBLEM

For sparsely connected traveling salesman graphs, the branch-and-bound algorithm may be applied without difficulty. On the other hand, for a maximally interconnected graph of n cities, difficulty may occur due to the (n-1)! different tours. In this case, however, there are on the order of n! connections. It could thus be more difficult to find the minimum gain connection to use as a bound (with gain appropriately defined) than to solve the traveling salesman problem itself. There is, however, a *lower bound* on the gain of the minimum gain connection which is in general higher than the bound on the minimum tour used in the algorithm and which is easily calculated. In fact, there are four such bounds suggested by the connection property of having one element from each row and one from each column. Let a zero element be selected from each row of the matrix. If each column contains exactly one selected entry, then a connection has been identified. If any column contains $M > 1$ selected elements, M-1 of these elements must be moved, incurring an increase in the length equal to the sum of the M-1 smallest row Δ's selected from the row Δ's of the M elements in the column. This process may be repeated for each multiple entry column and the increments added together. In Fig. 1f, the elements with dots in the upper right hand corner are selected. Column 6 is the only multiple entry column. The appropriate Δ's are 3, 2, and 3. Thus, the bound, the sum of the smallest two of these, is 5. Alternatively, attention may be focused on the columns with no elements selected (3 and 5). These may be filled at no cost only by moving the column 2 element. Columns 2, 3, and 5 have Δ's of 0, 4, and 2. If each column is to have exactly one entry, the bound may be increased by 2, the sum of the smallest two of these Δ's. The other two possible bounds are found by a symmetric process. One element per column is selected, such as those with dots in the lower left, Fig. 1f. Multiple element rows (4 and 6) may be eliminated, thereby increasing the bound by 2. Alternatively no-element rows (3 and 5) may be filled, thereby increasing the bound by 5. The maximum of the four bounds for this example is 5. Thus, a tighter bound of 56 + 5 = 61 is provided for the node $\overline{1\text{-}4}$ and the development of the remainder of the tree becomes unnecessary.

It should be mentioned that the tighter bound is used here to increase the bound beyond some limit such as the length of a previously found tour. Since an insufficient increase in the bound does not modify the branch-and-bound algorithm, and since some extra time is required to calculate each new bound, tighter bounds should be used only at "critical" points in the solution. Such useful points might occur after a new shortest tour has been found, or as a bound becomes fairly close to some minimal previously found tour length.

It is possible to estimate the number of tours represented by a particular node. The permanent of the connection matrix corresponding to the graph of the reduced matrix of a node can be evaluated to yield the number of connections represented by that node. This use of the permanent has been suggested by Jacob[5] for flow-graphs. The number of connections yields an upper bound on the number of tours which becomes more accurate as more branches are forbidden. If only the self-loops are forbidden, (n-1)! is the number of tours represented, where n is the order of the reduced matrix. In any case, no more than (n-1)! tours are possible. If a node represents very many tours, perhaps the calculation of bounds even more involved but larger than those suggested would be justified, whereas if very few tours are represented, one might as well continue with the branch-and-bound algorithm.

APPLICATION OF TREE REPRESENTATION TO FLOW-GRAPHS

Consider the system of equations $[A]\vec{X} - \vec{b} = 0$ and in particular the b vector and A matrix given in Fig. 2. How can the 18 connections and one-connections of this flow-graph, Fig. 2c, be found in a constructive and economical manner? The branch-and-bound solution tree of the traveling salesman problem may be adapted by neglecting to block subtours (multi-loop connections) and by branching on positive conditions only (always committing a matrix element, never prohibiting one). The line of the matrix with the most zeros is selected for branching, causing a node of the tree to branch to a new node for each non-zero element in the line. In Fig. 2a, a separate tree is developed for each of Δ, Δ^1, Δ^2

Δ^3, and Δ^4 respectively, representing the evaluation of the corresponding Δ's in near-optimal factored form. It is possible to calculate the values for all five determinants from these trees using just 36 multiplications (one for each numerical tree element not on the lowest level of the tree). In order to calculate each connection gain separately, 3 x 18 = 54 multiplications are required. The appropriate signs may be obtained either by affixing them by Laplacian expansion rules as the tree is developed ("-" superscript at nodes of trees) or by using Coates' loop rule to determine the sign of each connection as it is completed. In this "separate trees" method the resulting trees resemble well carried out Laplacian expansions of the corresponding determinants.

All connections and one-connections may be found in a single tree by taking the equation statement of Fig. 2 to be a 5 x 5 matrix including the literal row on top for bookkeeping purposes. It should be mentioned that signs may be correctly found by the Laplacian expansion rules for a 5 x 5 determinant. If some Δ is not desired, the corresponding literal of the top row is replaced by 0. It is desirable to require branching on rows only (rather than rows or columns) as well as branching on the top row last, so that the identifying literal appears on the last level of the tree, Fig. 2b. The number of multiplications required to evaluate all determinants from the single tree is reduced to 32, one for each numerical node of the tree except for the two on the top numerical level, rather than the 36 multiplications of the separate trees. Perhaps more significantly, the single tree requires much less effort to develop, which is reflected in the 34 numerical nodes (53 nodes total) vs. 54 numerical nodes (59 nodes total) in the separate trees expansion.

APPLICATION OF BOUNDING TO FLOW-GRAPHS

Bounding may be employed in the connection tree for the evaluation of a determinant. This procedure is useful if approximate values are sufficient. It is particularly effective if a large dynamic range is found in the matrix elements (cf. 100 to 1 in Fig. 3). A divisor, θ, is found for each row and column successively, to reduce the largest entry to 1. Such a matrix may be called reduced for flow-graphs. The product of the θ's thus forms an upper bound for the maximum connection gain. The row or column which contains the fewest 1's is selected for branching. Actually if there are many rows and columns with single 1's, Δ could be set to the next largest element for each row and column. Alternatively, Δ could be set to the number of "acceptable" elements i.e. those elements whose resulting bound would be high enough to warrant further branching, for each row or column. In either case, the row or column with smallest Δ would be selected for branching. In some cases it may be preferable to work with the log gain matrix rather than the gain matrix.

It is desirable to specify in the problem statement that all connections with gains larger than some number are to be found. This number may be reduced later in view of the resulting partial evaluation and error. In Fig. 3 the upper part of the solution tree represents all connections with gains $\geq 10^8$; the lower part is an extension for connections with gains $<10^8$ but $\geq 10^6$. The error in computing a partial tree may be found by summing up the number of connections represented, multiplied by the bound for each incomplete terminal node of the tree. The number of connections may be evaluated directly or by evaluating the appropriate permanent if many 0's are present in the matrix. It may be seen from the table of values and errors of Fig. 3, that the error bound is a very poor one. However, the effort to improve it would be greater than the effort to complete the tree, at least for small problems. In contrast, the approximate determinant values are quite good even though relatively few connections were developed. In this example, an actual error of less than 2% was found by evaluating only 21% of the connections or an error of $2 \times 10^{-4}\%$ by evaluating 46% of the connections. This remarkable accuracy occurred even though about half the matrix elements were large. As a rule of thumb, reducing the dynamic range of the matrix elements would require also reducing the proportion of large elements to obtain similar accuracy with the same proportion of the connections.

REFERENCES

1. Chien, R. T., "A Simplification of the Coates-Desoer Formula for the Gain of a Flow-Graph," Proc. IEEE, Sept. 1965, pp. 1240-1241.

2. Coates, C. L., "Flow-Graph Solutions of Linear Algebraic Equations," IRE Trans. on Circuit Theory, June 1959, pp. 170-187.

3. Davidson, E. S., "New Method of Solution of the Traveling Salesman Problem," Report No. 201, Dept. of Computer Sciences, University of Illinois, March 1966.

4. Desoer, C. A. "The Optimum Formula for the Gain of a Flow-Graph or a Simple Derivation of Coates' Formula," Proc. IRE, May 1960, pp. 883-889.

5. Jacob, J. P., "The Number of Terms in the General Gain Formulas for Coates and Mason Signal Flow-Graphs," IEEE Trans. on Circuit Theory, Dec. 1965, pp. 601-604.

6. Little, J. D. C., et al., "An Algorithm for the Traveling Salesman Problem," Operations Res., 1963, pp. 972-989.

	1	2	3	4	5	6
1	∞	27	28	16	30	24
2	7	∞	16	1	30	25
3	20	13	∞	2	5	0
4	21	16	23	∞	18	18
5	13	46	27	48	∞	5
6	23	5	5	9	5	∞

(a)

	1	2	3	4	5	6	θ	Δ
1	∞	11	12	0	14	8	16	8
2	1	∞	15	0	29	24	1	1
3	15	13	∞	2	5	0	0	2
4	0	0	7	∞	2	2	16	0
5	3	41	22	43	∞	0	5	3
6	13	0	0	4	0	∞	5	0
θ	5	0	0	0	0	0	48	
Δ	1	0	7	0	2	0		

(b)

1-4	1	2	3	5	6	Δ
2	0	∞	15	29	24	15
3	14	13	∞	5	0	5
4	∞	0	7	2	2	2
5	2	41	22	∞	0	2
6	12	0	0	0	∞	0
θ	1	0	0	0	0	1
Δ	2	0	7	2	0	

(c)

1-4 2-1	2	3	5	6	θ	Δ
3	13	∞	5	0	0	5
4	∞	5	0	0	2	0
5	41	22	∞	0	0	22
6	0	0	0	∞	0	0
Δ	13	5	0	0	2	

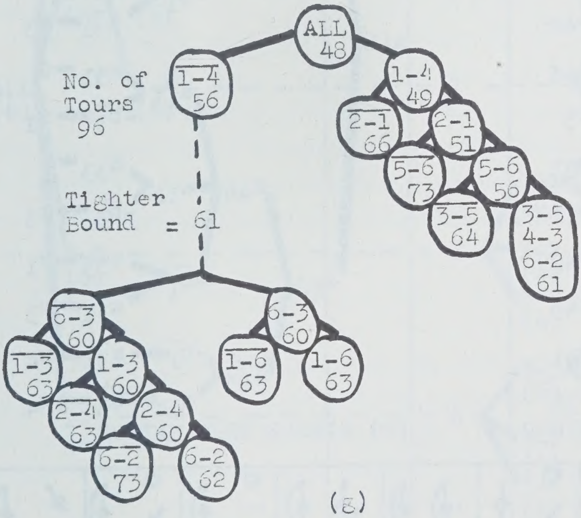
(d)

1-4 2-1 5-6	2	3	5	θ	Δ
3	8	∞	0	5	8
4	∞	5	0	0	5
6	0	0	∞	0	0
Δ	8	5	0	5	

(e)

$\overline{1\text{-}4}$	1	2	3	4	5	6	θ	Δ
1	∞	3	4	∞	6	0	8	3
2	1	∞	15	0	29	24	0	1
3	15	13	∞	2	5	0	0	2
4	0	0	7	∞	2	2	0	0
5	3	41	22	43	∞	0	0	3
6	13	0	0	4	0	∞	0	0
Δ	1	0	4	2	2	0	8	

(f)

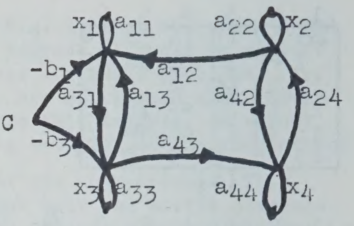

ALL 48
$\overline{1\text{-}4}$ 56
No. of Tours 96
Tighter Bound = 61
1-4 49
$\overline{2\text{-}1}$ 66
2-1 51
$\overline{5\text{-}6}$ 73
5-6 56
$\overline{3\text{-}5}$ 64
3-5 4-3 6-2 61
$\overline{6\text{-}3}$ 60
6-3 60
$\overline{1\text{-}3}$ 63
1-3 60
$\overline{1\text{-}6}$ 63
1-6 63
$\overline{2\text{-}4}$ 63
2-4 60
$\overline{6\text{-}2}$ 73
6-2 62

(g)

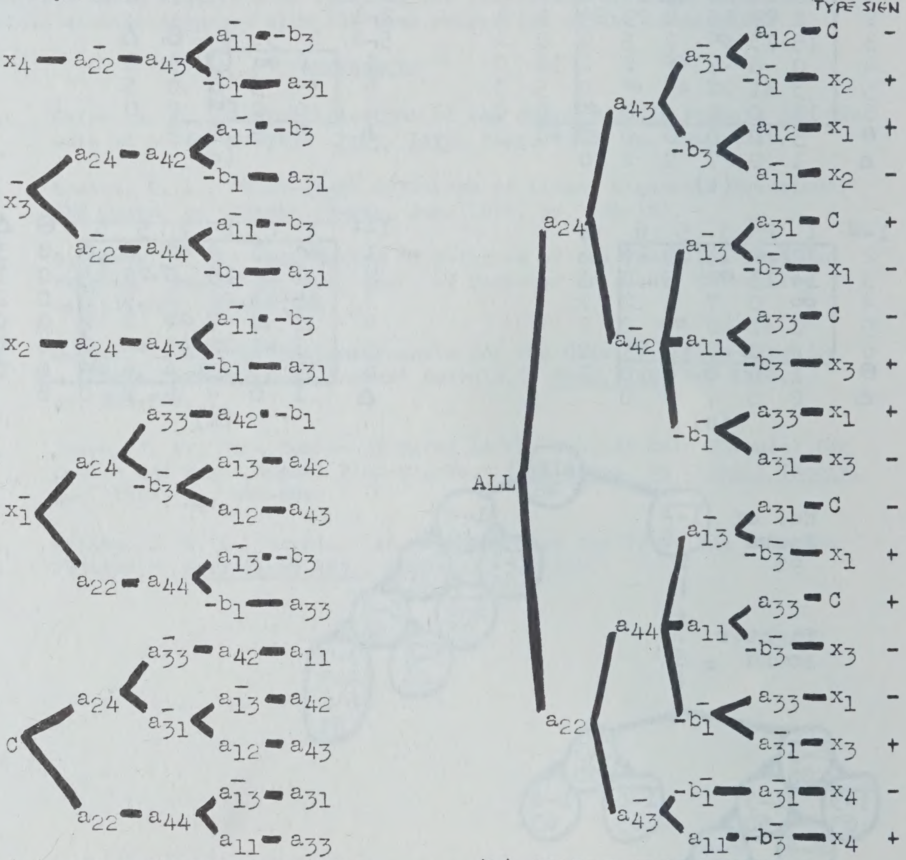
Fig. 1 TRAVELING SALESMAN PROBLEM
(TIGHTER BOUNDING)

$$\begin{array}{c|cccc}
C & x_1 & x_2 & x_3 & x_4 \\
\hline
-b_1 & a_{11} & a_{12} & a_{13} & 0 \\
0 & 0 & a_{22} & 0 & a_{24} \\
-b_3 & a_{31} & 0 & a_{33} & 0 \\
0 & 0 & a_{42} & a_{43} & a_{44}
\end{array}$$

EQUATION MATRIX

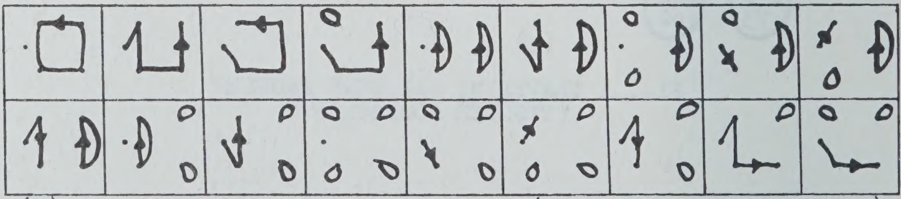

x_1
a_{11}
a_{22}
x_2
$-b_1$
a_{12}
a_{31}
a_{13}
a_{42}
a_{24}
C
$-b_3$
a_{43}
x_3
a_{33}
a_{44}
x_4

FLOW-GRAPH


CONN. TYPE SIGN
$x_4 - \bar{a}_{22} - a_{43} < \bar{a}_{11} \cdot -b_3$ / $-b_1 - a_{31}$
$\bar{x}_3$
$a_{24} - a_{42} < \bar{a}_{11} \cdot -b_3$ / $-b_1 - a_{31}$
$\bar{a}_{22} - a_{44} < \bar{a}_{11} \cdot -b_3$ / $-b_1 - a_{31}$
$x_2 - a_{24} - a_{43} < \bar{a}_{11} \cdot -b_3$ / $-b_1 - a_{31}$
$\bar{x}_1$
a_{24}
$\bar{a}_{33} - a_{42} \cdot -b_1$
$-\bar{b}_3 < \bar{a}_{13} - a_{42}$ / $a_{12} - a_{43}$
$a_{22} - a_{44} < \bar{a}_{13} \cdot -b_3$ / $-b_1 - a_{33}$
C
a_{24}
$\bar{a}_{33} - a_{42} - a_{11}$
$\bar{a}_{31} < \bar{a}_{13} - a_{42}$ / $a_{12} - a_{43}$
$a_{22} - a_{44} < \bar{a}_{13} - a_{31}$ / $a_{11} - a_{33}$
ALL
a_{24}
a_{43}
$\bar{a}_{31} < a_{12} - C$ − / $-\bar{b}_1 - x_2$ +
$-b_3 < a_{12} - x_1$ + / $\bar{a}_{11} - x_2$ −
$\bar{a}_{42}$
$\bar{a}_{13} < a_{31} - C$ + / $-\bar{b}_3 - x_1$ −
$a_{11} < a_{33} - C$ − / $-\bar{b}_3 - x_3$ +
$-\bar{b}_1 < a_{33} - x_1$ + / $\bar{a}_{31} - x_3$ −
a_{22}
a_{44}
$\bar{a}_{13} < a_{31} - C$ − / $-\bar{b}_3 - x_1$ +
$a_{11} < a_{33} - C$ + / $-\bar{b}_3 - x_3$ −
$-\bar{b}_1 < a_{33} - x_1$ − / $\bar{a}_{31} - x_3$ +
$\bar{a}_{43} < -\bar{b}_1 - \bar{a}_{31} - x_4$ − / $a_{11} \cdot -\bar{b}_3 - x_4$ +

(a) SEPARATE TREES METHOD

(b) SINGLE TREE METHOD



(c) CONNECTIONS AND ONE-CONNECTIONS (FROM SINGLE TREE METHOD)

Fig. 2 TREE REPRESENTATIONS OF CONNECTIONS

$$\begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \\ \hline 2 & 2 & 2 & 0 \\ 0 & 2 & 0 & 2 \\ 2 & 0 & 2 & 0 \\ 0 & 2 & 2 & 2 \end{array}$$

LOG GAIN
MATRIX

CONN. GAIN LIMIT	VALUE ($\times 10^{-8}$)	NO. OF TREE NODES	NO. OF CONNECTIONS
$\geq 10^8$	$-1 \pm .19$	24	5
$\geq 10^6$	$-.98 \pm .0013$	44	11
ALL	$-.98000199$	65	24

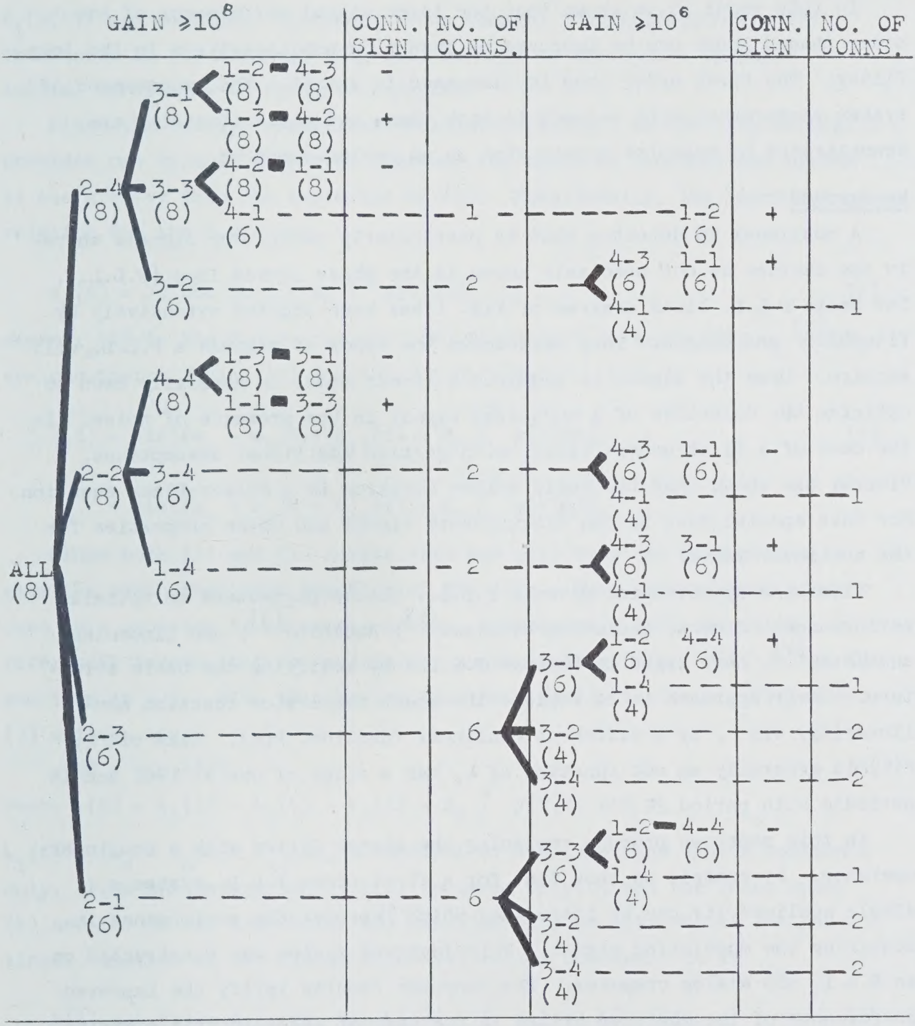

GAIN ≥10^8
CONN. SIGN
NO. OF CONNS.
GAIN ≥10^6
CONN. SIGN
NO. OF CONNS.
ALL (8)
2-4 (8)
3-1 (8)
1-2 (8) 4-3 (8) -
1-3 (8) 4-2 (8) +
3-3 (8)
4-2 (8) 1-1 (8) -
4-1 (6)
1
1-2 (6) +
3-2 (6)
2
4-3 (6) 1-1 (6) +
4-1 (4)
1
2-2 (8)
4-4 (8)
1-3 (8) 3-1 (8) -
1-1 (8) 3-3 (8) +
3-4 (6)
2
4-3 (6) 1-1 (6) -
4-1 (4)
1
1-4 (6)
2
4-3 (6) 3-1 (6) +
4-1 (4)
1
2-3 (6)
6
3-1 (6)
1-2 (6) 4-4 (6) +
1-4 (4)
1
3-2 (4)
2
3-4 (4)
2
2-1 (6)
6
3-3 (6)
1-2 (6) 4-4 (6) -
1-4 (4)
1
3-2 (4)
2
3-4 (4)
2

Fig. 3 BOUNDING IN FLOW-GRAPHS

IMPROVEMENT OF PHASE-LOCKED LOOP PERFORMANCE BY THE INTRODUCTION OF NONLINEARITIES

L. L. CLELAND AND B. J. LEON
Purdue University
Lafayette, Indiana 47907

ABSTRACT

In this paper it is shown that the large signal performance of low order phased loops can be improved by inserting nonlinearities in the loop filter. The first order loop is discussed in detail. The improvement of system performance with respect to lock range and phase linearity are demonstrated by examples constructed on an analog computer.

Background

A nonlinear FM detector that is particularly useful for signals wherein the carrier is not precisely known is the phase locked loop (P.L.L.). The basic P.L.L. block diagram of Fig. 1 has been studied extensively by Viterbi[1] and others. They have shown the types of signals a P.L.L. will acquire. Once the signal is acquired a linear model is generally used to optimize the detection of a modulated signal in the presence of noise. In the case of a first order filter, with certain additional assumptions, Viterbi has shown that the basic system equation is a Folker-Plank equation. For this special case he can discuss both signal and noise properties for the nonlinear model.

Viterbi's approach is, given a P.L.L., choose parameters to optimize performance. Others, including Williams[2], Robinson[3], and Lindenlaub and Uhran[4], have tried to improve P.L.L.s by modifying the basic structure. Their approach is to replace the phase comparator function nonlinearity, $\sin \lambda$, by a different nonlinear function, $F(\lambda)$. Like $\sin \lambda$, $F(\lambda)$ is generally an odd function of λ, has a slope of one at $\lambda=0$, and is periodic with period 2π.

In this paper we propose replacing the linear filter with a nonlinear operator. By example we show that for a first order P.L.L. system a simple nonlinearity can be introduced which improves the performance in acquiring the modulating signal. This improved system was constructed on an E.A.I. 680 analog computer. The computer results verify the improved performance of the modified system in the case of deterministic signals.

Loop Modification by Adding Nonlinearities

The basic modification of the P.L.L. of Fig. 1 proposed in this paper is the replacement of the linear filter with a set of nonlinear integrators. The general functional form -- a form readily implimented in practice for low frequency signal processing -- is

$$y(t) = K_1\{A_o(x(t)) + \int^t A_1(x(t_1)dt_1 + \int^t\int^{t_1} A_2(x(t_2))dt_2dt_1 \tag{1}$$

$$+ \ldots + \int^t\int^{t_1} \ldots \int^{t_{n-1}} A_n(x(t_n))dt_n dt_{n-1} \ldots dt_1\}$$

where $y(t)$ is the output, $x(t)$ is the input, K_1 is a constant, and the $A_i[x(t)]$ are nonlinear functions to be chosen. The integrals are left indefinite in (1) although they will be made to accomodate definite initial conditions in specific examples.

Since the voltage controlled oscillator (V.C.O.) in the loop of Fig. 1 provides one dynamic degree of freedom, the order of the modified system is one greater than the subscript in (1). Specifically, the input-output relation for the V.C.O. is

$$x_o(t) = \sqrt{2} \cos [\omega_o t + K_2 \int^t y(\tau)d\tau] \tag{2}$$

where $x_o(t)$ is the V.C.O. output, $y(t)$ is the V.C.O. input, and ω_o and K_2 are constants. Using (2) for the V.C.O. output

$$x(t) = \sin[(\omega_c - \omega_o)t + \lambda_{i1}(t) - K_2 \int^t y(\tau)d\tau] \tag{3}$$

$$+ \sin[(\omega_c + \omega_o)t + \lambda_{i1}(t) + K_2 \int^t y(\tau)d\tau]$$

Since both (3) and (1) relate $x(t)$ and $y(t)$ they can be combined to give the basic nonlinear equation of the loop. Even in the first order case this equation is extremely messy. Consequently, all the cited references assume that the multiplyer and/or system filter will reject the sum term in (3). With this assumption the notation can be simplified and (3) written as

$$x(t) = \sin \lambda(t) \tag{4}$$

where $\lambda(t) = \lambda_i(t) - \lambda_o(t) = \lambda_i(t) - K_2 \int^t y(\tau)d\tau$ and $\lambda_i(t) = (\omega_c - \omega_o)t + \lambda_{i1}(t)$, a function of the input and system constants only. From the definition of $\lambda(t)$ in terms of $y(t)$ and the input under (4) and the filter expression (1) relating $y(t)$ to $x(t)$ we get the nonlinear equation of the system with $\lambda(t)$ as the unknown. It is

$$\lambda(t) = \lambda_i(t) - K \int^t \{A_o \sin \lambda(\tau)) + \int^\tau A_1(\sin \lambda(t_1))dt_1 \tag{5}$$

$$+ \ldots + \int^\tau\int^{t_1} \ldots \int^{t_{n-1}} A_n(\sin \lambda(t_n))dt_n \ldots dt_1\}d\tau$$

where $K = K_1 K_2$.

A more conventional feedback model for (5) is shown in Fig. 2. The sine function, which is the nonlinear consequence of the multiplier in Fig. 1, is called the phase comparator. In the discussion that follows, any other phase comparator function could be used. Thus in subsequent work we use a general $F(\lambda)$ to designate phase comparator.*

The First Order Case

In the first order case (5) becomes

$$\lambda(t) = \lambda_i(t) - K \int^t A_o\{F[\lambda(\tau)]\}d\tau. \tag{6}$$

A standard test for a P.L.L. is to start from a locked condition at t=0 and make a step frequency change at the input. Thus $\lambda_i(t) = \Delta\omega t + \theta$, and equation (6) is

$$\Delta\omega = \frac{d\lambda(t)}{dt} + K A_o\{F[\lambda(t)]\}. \tag{7}$$

The system will lock on the new frequency if $\frac{d\lambda(t)}{dt}$ goes to zero. Since the system is assumed in lock at t=0, the initial condition is $\lambda(0)=0$ and $K A_o(F(0)) = 0$. Thus when $\Delta\omega$ in (7) is positive, $\lambda(t)$ is increasing from zero at t=0. To slow this increase $K A_o(F(\lambda(t)))$ must be positive for small positive $\lambda(t)$. Furthermore, if $\frac{dx}{dt}$ is to approach zero for a particular $\Delta\omega$ the range of the composite function $K A_o(F)$ must include that $\Delta\omega$.

When A_o is linear and F is the sine function there is a maximum $\Delta\omega$ that the loop can lock to. In a P.L.L. with such a maximum, this maximum is called the lock range (LR). For the first order loop of (7) the LR is the maximum of the range of $K A_o(F)$.

A second important criterion for a P.L.L. is the synchronization time T_s. This quantity is defined as the time required for the loop to respond to a step input so that $\lambda(t)$ stays within 10% of its final value. For the first order loop (7) can be converted to an integral expression for T_s. It is

$$T_s = \int_0^{0.9\lambda_f} \frac{d\lambda}{\Delta\omega - K A_o(F(\lambda))} \tag{8}$$

where λ_f is the final value of λ for the given $\Delta\omega$ input. Clearly if $K A_o(F(\lambda))$ increases more rapidly with λ, T_s will be smaller. The examples below show how A_o can be chosen so that LR is increased and/or T_s decreased.

---

*We could also combine the zeroth order term of the nonlinear filter $A_o(F(\lambda))$, with the phase comparator to give a more complex function of λ. For simplicity we keep them separate.

Improvement of a Conventional Loop

The system designer has complete freedom in specifying the composit function, $a_o(\lambda) = A_o(F(\lambda))$, over the domain of $F(\lambda)$ for which the monotonic range corresponds to negative feedback in the loop. To extend the L.R., $a_o(\lambda)$ should be monotonic with a larger range than that of $F(\lambda)$. For a fixed loop gain over a large phase range, $a_o(\lambda)$ should be linear for the monotonic range of $F(\lambda)$ corresponding to negative feedback. Assuming both phase linearity and lock range are important design criterion, then the ideal composit function for a simple P.L.L. should be linear for as much of one cycle as possible and then head toward $\pm\infty$ at $\pm\frac{\pi}{2}$ respectively. One possibility is

$$a_o(\lambda) = \begin{cases} \lambda & , \text{ for } -(\pi/2-\epsilon) \leq \lambda \leq (\pi/2-\epsilon) \\ \dfrac{-1}{\lambda-\pi/2} + \pi/2 - \epsilon - 1/\epsilon, & \text{for } (\pi/2-\epsilon) \leq \lambda < \pi/2 \\ \dfrac{1}{\lambda+\pi/2} - \pi/2 + \epsilon + 1/\epsilon, & \text{for } -(\pi/2-\epsilon) \geq \lambda > -\pi/2 \end{cases} \tag{9}$$

We cannot construct a nonlinear filter such that $a_o(\lambda)$ goes to ∞ at $\lambda = \pi/2$. We can realize a filter with $a_o(\pi/2)$ very large. A realizable function which adequately approximates (9) is

$$a_o(\lambda) = \begin{cases} \lambda & , -\eta\,\pi/2 \leq \lambda \leq \eta\,\pi/2 \\ M[\sin\lambda - \sin(\eta\,\pi/2)] + \eta\,\pi/2, & \eta\,\pi/2 < \lambda \leq \pi/2 \\ M[\sin\lambda - \sin(\eta\,\pi/2)] - \eta\,\pi/2, & -\eta\,\pi/2 > \lambda \geq \pi/2 \end{cases} \tag{10}$$

where $M \gg 1$ and $\eta < 1$. If η is selected such that $\sin \eta\,\pi/2$ equals, for example, 0.9, then the linear phase range is $\eta\pi$ radians. Assuming the slope of $A_o(F)$ is not restricted then M can be selected on the basis of the V.C.O. dynamic range. The extent to which the lock range has been improved is $a_o(\pi/2)$.

Experimental Verification

On the basis of the above arguments a simple P.L.L. system was constructed on an EAI analog computer. A simplified block diagram of the experimental system is shown in Fig. 3. The system with and without the nonlinearity (D.F.G.) could be compared by switching S1 and properly adjusting gain pot. RT. Wavetek* number one was used as the system V.C.O., Wavetek number two represented the input signal, and Wavetek number three was used to frequency step the input signal. Two low pass filters, A_3 and A_4, were used to attenuate the sum term output of the multiplier such that the V.C.O. responded only to the difference term. The poles of these filters were located well above the system response range yet below the 900Hz V.C.O. center frequency. The loop gain was chosen to allow a normal lock range (without the D.F.G.) of 44Hz, i.e., from 878Hz to 922Hz. For $\eta = 7/9$ the expected linear phase range was $7/9\pi$. M was chosen such that $a_o(\pi/2) = 3.4$; i.e., M = 36.4. Thus the lock range should be extended to 149.6Hz.

---

*Wavetek Model 111 function generators with a V.C.O. feature were used.

One quadrant of the experimental composit function is shown in Fig. 4. A considerable improvement in phase linearity is noticeable. The improvement in the synchronization time over the nonlinear range of sin λ is shown in Fig. 5. Typical step response curves for the two systems are shown in Fig. 6. The experimental lock range was from 835Hz to 967Hz or $\Delta f = 132$Hz. This is in good agreement with the predicted value of $\Delta f = 149.6$Hz.

Improvement of a Tanlock P.L.L.

A Tanlock system [3] has a phase comparator function

$$F(\lambda) = \frac{(1 + k)^m \sin \lambda(t)}{[1 + k \cos \lambda(t)]^m} \tag{11}$$

where m is an integer and $k < 1$. A simple Tanlock system with m = 1 and k = 0.75 was constructed on the EAI computer. A simplified block diagram of the experimental Tanlock system is shown in Fig. 7. $\eta\lambda_{max}$ was chosen as $\frac{11}{18}\pi$ and M was chosen as 7.8. One quadrant of $F(\lambda)$ and $a_o(\lambda)$ for the experimental systems are shown in Fig. 8. The loop gain was chosen to allow a normal lock range (without the D.F.G.) of 44Hz as before. The theoretical lock range should be extended to 88Hz. The experimental lock range was from 858Hz to 943 or $\Delta f = 85$Hz. The synchronization times for the two systems were nearly the same for all values of $\lambda < \eta\,\lambda_{max}$ as expected due to the small differences between $F(\lambda)$ and $a_o(\lambda)$ in this range.

The Second Order Case

In the second order case (5) becomes

$$\lambda(t) = \lambda_i(t) - K \int^t A_o\{F[\lambda(\tau)]\}\, d\tau$$
$$- K \int^t \int^\tau A_1\{F[\lambda(t_1)]\}\, dt_1 d\tau. \tag{12}$$

It can be shown that for step frequency inputs, $\lambda_i = \Delta\omega t + \theta$, the lock range is infinite. For a linear frequency input, $\lambda_i = 1/2\, Rt^2 + \Delta\omega t + \theta$, the frequency rate, R, is one of the more important design criterion. By differentiating (12) can be written

$$R = \frac{d^2\lambda(t)}{dt^2}\, K \frac{d\, A_o[F(\lambda)]}{d\lambda}\, \frac{d\lambda}{dt} + K\, A_1\{F[\lambda(t)]\}. \tag{13}$$

Again $\frac{d\lambda}{dt}$ must be zero when the system is in lock. Therefore, R_{max} is specified by the maximum value of $KA_1\{F[\lambda(t)]\}$. The desired function, $A_1\{F[\lambda(t)]\}$ is of the same type as was needed to improve the L.R. for the first order loop. Hence, one nonlinearity ahead of a linear loop filter will indeed improve the rate range.

BIBLIOGRAPHY

[1] A.J. Viterbi, Principles of Coherent Communications, McGraw-Hill Book Co., 1966.

[2] W.J. Williams, "Selection of Phase Sensitive Detector for Space Radar," IEEE Region Six Conference, Salt Lake City, Utah, April, 1964.

[3] L.M. Robinson, "Tanlock: A P.L.L. of Extended Tracking Capability," Proc. National Winter Conference on Military Electronics, Los Angeles, California, pp. 396-421, February, 1962.

[4] J.C. Lindenlaub and J.J. Uhran, "Threshold Study of P.L.L. Systems," Purdue University, School of Electrical Engineering, Electronics Systems Research Laboratory, Technical Report TR-EE66-19, December, 1966.

ACKNOWLEDGEMENT

The research reported herein was supported by the National Science Foundation under grant GK-01235. The authors are indebted to Stephen A. Malott for his efforts in obtaining the analog computer data presented in this paper.

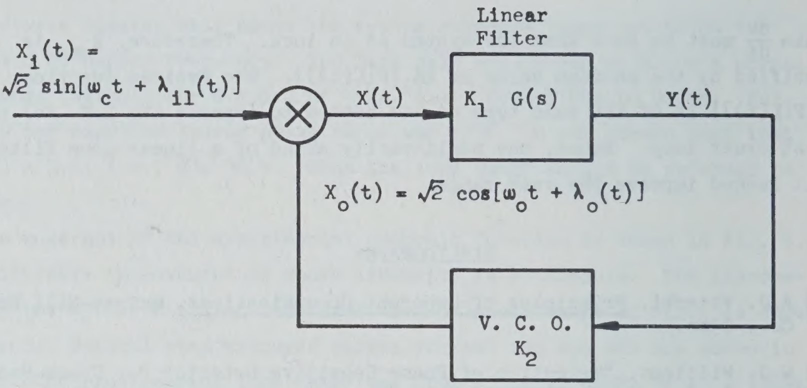

Linear Filter
$X_1(t) = \sqrt{2}\sin[\omega_c t + \lambda_{i1}(t)]$
$X(t)$
$K_1 \quad G(s)$
$Y(t)$
$X_o(t) = \sqrt{2}\cos[\omega_o t + \lambda_o(t)]$
V. C. O.
K_2

Figure 1: Basic P. L. L. Block Diagram

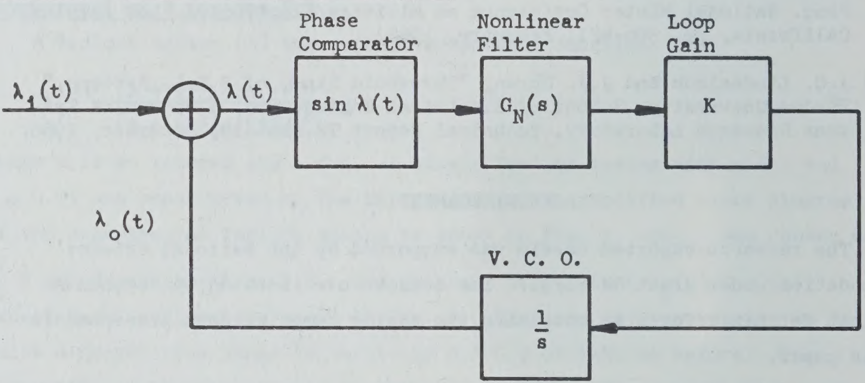

Phase Comparator
Nonlinear Filter
Loop Gain
$\lambda_i(t)$
$\lambda(t)$
$\sin\lambda(t)$
$G_N(s)$
K
$\lambda_o(t)$
V. C. O.
$\frac{1}{s}$

Figure 2: Modified P. L. L.

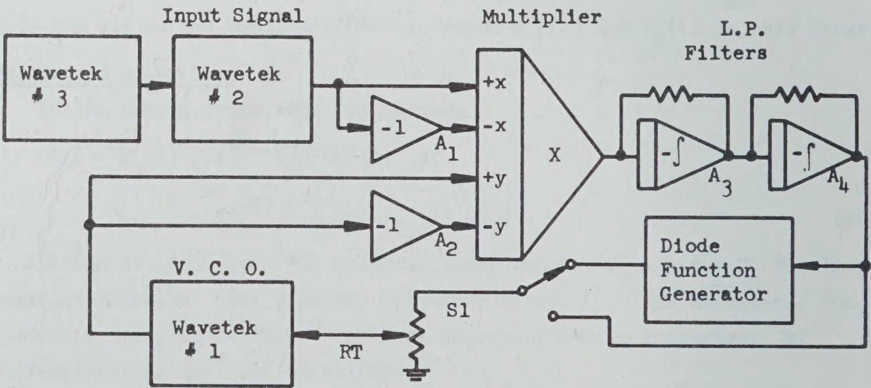

Input Signal
Multiplier
L.P. Filters
Wavetek # 3
Wavetek # 2
-1
A_1
+x
-x
+y
-y
X
-1
A_2
$-\int$
A_3
$-\int$
A_4
Diode Function Generator
V. C. O.
Wavetek # 1
RT
S1

Figure 3: Experimental Simple P. L. L.

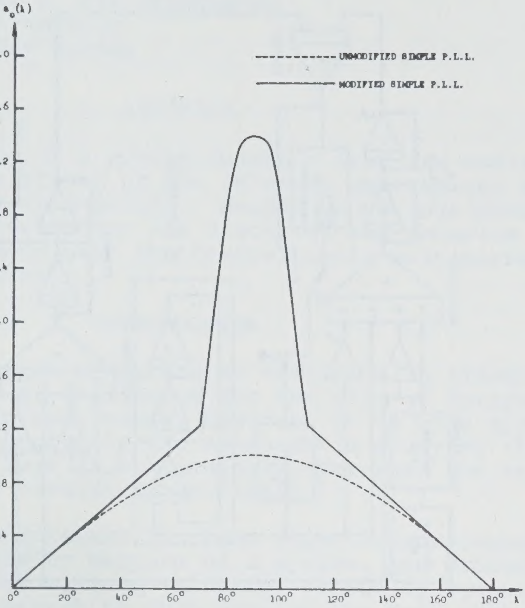


Figure 4: Composit Function for Simple P.L.L.

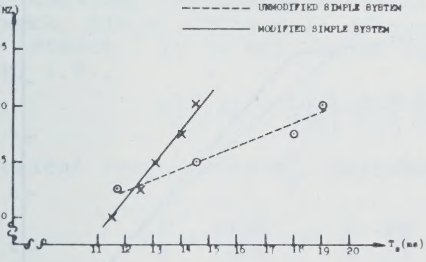


Figure 5: Synchronization Time for Simple P.L.L. with and without Nonlinearity.

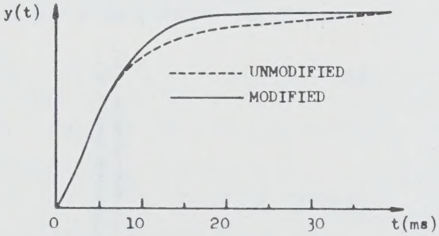


Figure 6: Typical Step Response for Simple P. L. L. ($\Delta f = 20H_z$).

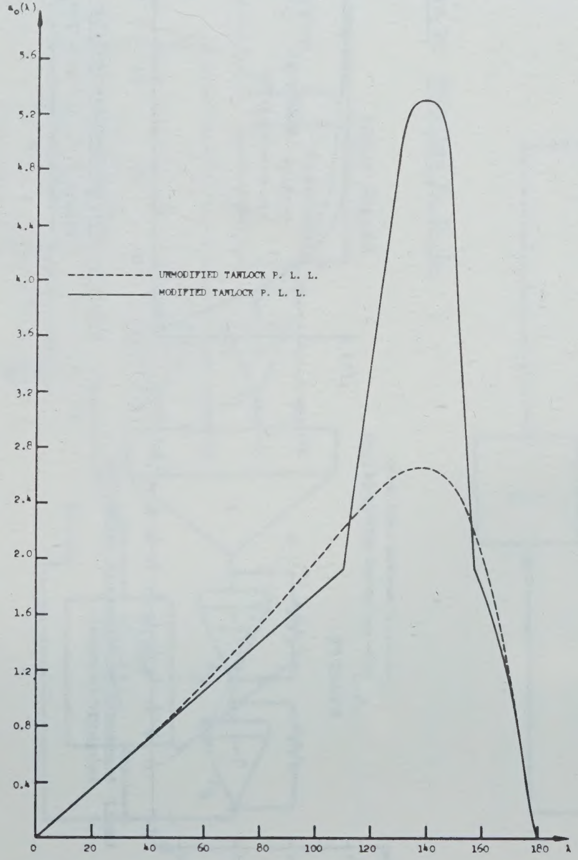


FIGURE 8: COMPOSIT FUNCTION FOR TANLOCK P.L.L.

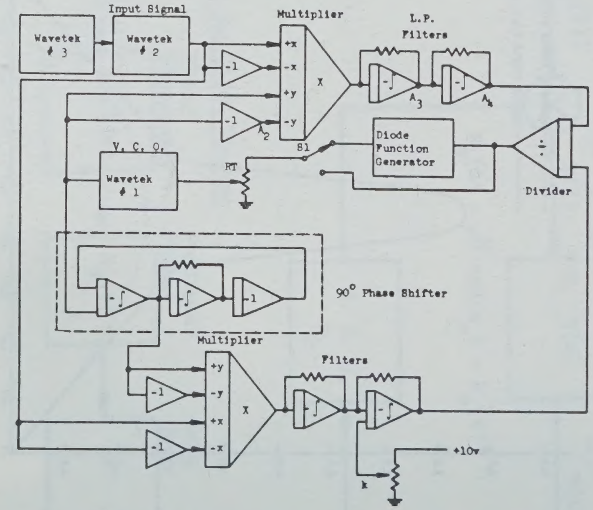

Input Signal
Wavetek # 3
Wavetek # 2
Multiplier
+x
-x
+y
-y
X
-1
L.P. Filters
-∫
A_3
A_4
A_2
Diode Function Generator
S1
RT
V. C. O.
Wavetek # 1
÷
Divider
90° Phase Shifter
Multiplier
Filters
+10v
k

Figure 7: EXPERIMENTAL TANLOCK P.L.L.

A PROBABILISTIC APPROACH TO THE PARAMETER SENSITIVITY PROBLEM

B. LIU and M. E. VAN VALKENBURG
Princeton University
Princeton, New Jersey

ABSTRACT

The deviation of a system function from its nominal value due to random variations or the inherent uncertainty of the system parameters is examined. Bounds on various probabilistic measures of this deviation are discussed and examples are given. A comparison is made with the deterministic multiparameter sensitivity function.

I. INTRODUCTION

The significance of parameter variation in linear networks and systems is well known for the case of deterministic models. [1-7]. In many cases, however, it is more appropriate to make use of a probabilistic approach in studying the sensitivity problem. This is particularly true when the number of parameters in the network is not small.

Considerable interest has been shown in worst-case variation as a sensitivity measure of a system, and procedures have been given for maximum (minimum) search by computer [8]. An alternative approach to this problem has been that of deriving bounds of the variations [9]. However, the bounds derived so far tend to be pessimistic [8]. We give here a simple way of calculating the approximate worst-case performance when the variation of parameters is not large. A comparison is made of this variation with the results obtained using the probabilistic approach.

II. SINGLE PARAMETER CASE

Consider the system function which describes a linear network, $H(\omega,x)$ where ω is frequency and x is an element in the network. It is well known that H is a bilinear function of x, i.e.,

$$H(\omega,x) = \frac{A(\omega) + x\,B(\omega)}{C(\omega) + x\,D(\omega)} \tag{1}$$

Classical sensitivity S_x^H, defined by

$$S_x^H = \frac{\partial \ell n H}{\partial \ell n x} = \frac{x}{H}\,\frac{\partial H}{\partial x} \tag{2}$$

has been used in the study of the change of system performance with respect to parameter variation. The real and imaginary parts of S_x^H are related separately to the amplitude and phase variations of H caused by a small change Δx in x. The first-order effects are given by

$$\frac{\Delta|H|}{|H|} \approx \operatorname{Re} S_x^H \,\frac{\Delta x}{x} \tag{3}$$

and

$$\Delta \arg H \approx \text{Im } S_x^H \frac{\Delta x}{x} \tag{4}$$

where $|H|$ and $\Delta|H|$ and $\Delta\arg H$ denote respectively the magnitude of H, the change in $|H|$, and the change in the argument of H.

Traditionally, the problem of system sensitivity and parameter variation has been treated by using a deterministic approach by assuming the variation Δx is a fixed quantity. However, strictly speaking, the element values in a system are only known probabilistically in view of inherent uncertainties. In addition, our knowledge of the variation of the variation of these element values due to aging, environment changes, etc. is conditioned by a certain degree of incompleteness. It is therefore more reasonable to take x as a random variable whose mean $\bar{x}$ is regarded as its nominal value and whose variance σ_x^2 gives a measure of its inherent inaccuracy.

At a fixed ω, the expected value of $H(\omega,x)$ as given in Eq. 1 is given by

$$E\{H(\omega,x)\} = \int \frac{A(\omega) + xB(\omega)}{C(\omega) + xD(\omega)}\, p(x)\, dx \tag{5}$$

where E denotes the expectation, and $p(x)$ is the probability density of x. Since one normally takes

$$H(\omega,\bar{x}) = \frac{A(\omega) + \bar{x}B(\omega)}{C(\omega) + \bar{x}D(\omega)} \tag{6}$$

as the nominal H, it is of interest to compare the two nominal values of H as given by Eqs. 5 and 6.

We may write x as $\bar{x} + \Delta$, where the random variable Δ has zero mean and variance σ_x^2. If σ_x is small, $H(\omega,x)$ may be expaned in a power series

$$H(\omega,x) = H(\omega,\bar{x}) + \frac{\partial H}{\partial x}\Delta + \frac{1}{2}\frac{\partial^2 H}{\partial x^2}\Delta^2 + \dots \tag{7}$$

and the first few terms can be taken as a good approximation. Since $E\{\Delta\} = 0$, it can be seen that the difference between $E\{H(\omega,x)\}$ and $H(\omega,x)$ is a second-order effect.

This result can also be seen in terms of a graphical representation. The bilinear Eq. 1 maps the real x line into a circle for a fixed ω as illustrated in Fig. 1(a). Suppose x deviates from its nominal value $\bar{x}$ by a fixed δ. Then $H(\omega,\bar{x} + \delta)$ should be on the circle as shown in the slightly expanded illustration about $H(\omega,\bar{x})$ of Fig. 1(b). The approximation, $H(\omega,\bar{x}) + \frac{\partial H}{\partial x}\delta$, obtained by taking only the first two terms in the series expansion, Eq. 7, lies on the tangent to the circle at $H(\omega,\bar{x})$. Similarly, $H(\omega,\bar{x}-\delta)$ is on the circle and $H(\omega,\bar{x}) - \frac{\partial H}{\partial x}\delta$ is on the tangent. Since the approximation lies on a straight line and since the random deviation is assumed to have zero mean, the resulting approximation to $E\{H(\omega,x)\}$ coincides with $H(\omega,\bar{x})$. The actual $E\{H(\omega,x)\}$ would lie somewhere inside the circle as shown. By taking more terms in the power series, the approximation to the expected value of H would approach its true value.

If the exact distribution of x is known, or a reasonable one can be assumed, then $E\{H(\omega,x)\}$ can be determined, though in most cases numerical calculation by a computer is required. For example, if x is gaussian with mean $\bar{x}$ and variance σ_x^2, Eq. 5 can be written as[1]

$$E\{H(\omega,x)\} = \frac{B}{D} + \left(\frac{E}{D} - \frac{BF}{D^2}\right) \frac{j}{\sigma_x} \sqrt{\frac{\pi}{2}}\, W\left(\frac{-F}{\sqrt{2}\sigma_x D}\right) \tag{8}$$

where $E = A+\bar{x}B$, $F = C+\bar{x}D$ and the function $W(z)$ is related to the complementary error function by [10] $W(z) = e^{-z^2}\operatorname{erfc}(-iz)$ If x is uniformly distributed between $\bar{x} \pm \sqrt{3}\,\sigma_x$, the result is

$$E\{H(\omega,x)\} = \frac{B}{D} + \frac{ED - BF}{2\sqrt{3}\sigma_x D} \ell n \frac{F + \sqrt{3}\sigma_x D}{F - \sqrt{3}\sigma_x D} \tag{9}$$

Provided C/D is not real.

III. MULTIPLE PARAMETER CASE

Let x_1 and x_2 be two elements in a linear network. The system function $H(\omega, x_1, x_2)$ is then bilinear in each of x_1 and x_2. We take x_1 and x_2 to be independent random variables with means $\bar{x}_1$ and $\bar{x}_2$ and variances σ_1^2 and σ_2^2. Again, $H(\omega, \bar{x}_1, \bar{x}_2)$ is usually taken to be the nominal H and it can be argued as before that $H(\omega, \bar{x}_1, \bar{x}_2)$ is a good approximation of the actual mean $E\{H(\omega, x_1, x_2)\}$.

At each ω, and for each fixed x_2, the locus of H as a function of x_1 is a circle, and for each x_1, the locus of H as a function of x_2 is also a circle. Suppose the value of x_1 is to vary between $\bar{x}_1 \pm \delta_1$ and the range of x_2 is $\bar{x}_2 \pm \delta_2$, then the variation of $H(\omega, x_1, x_2)$ is as illustrated in Fig. 2. The value of H is then limited to the region bounded by the four dashed circular arcs. If none of these four arcs is tangent to circles with center at the original, or to straight lines passing through the origin, it is seen that the extremes of $|H|$ and arg H can occur only at the "corner points," a,b,c, or d. Even when this is not true because some of the arcs are tangent to circles with center at origin or to straight lines through the origin, the extreme values evaluated at the corners are still very good approximations to the true extremes. The above argument can be extended to situations with more than two parameters. Figure 3 illustrates a 3-parameter case.

---

[1] For simplicity, the argument ω is sometimes omitted.

The significance of the problem of worst-case variation has been recognized by many authors. Most of the results in the past have concentrated along two lines. The first is to device bounds which usually turn out to be rather pessimistic. The second is to actually search for the extremes in the parameter space and usually requires considerable computational effort. The observation made above can be used to determine approximately the worst case variation of H by simply evaluating H at all of the corner points and comparing the results. Thus for a network of n parameters, x_i with range $\bar{x}_i \pm \delta_i$, $i = 1,2,\ldots n$, one needs to calculate H for each of the 2^n combinations of the parameter values, $x_i = \bar{x}_i \pm \delta_i$. The extremums obtained in this manner can be taken as approximations of the actually extremums.

As an illustration, consider the third-order Chebyshev low-pass filter shown in Fig. 4. The sensitivity problem of this network has been studied by Kelly [8] who calculated the actual maximum and minimum of the amplitude characteristic. The result of calculation based upon the approach outlined above is shown in Fig. 5 for ±5% and ±10% variations of all elements. Also shown is the actual maximum and minimum as calculated by Kelly. For this order of parameter variation, it is seen that our method gives reasonably good approximations except near the cut off frequency where the change is fast. However, if the range of variation is increased, one should expect that our approximation procedures will not give reliable results.

IV. MULTIPARAMETER SENSITIVITY AND STATISTICAL VARIATION

Let x_i, $i = 1,2,\ldots n$ be element values. The first order approximation to the variation in the system function $H(\omega,x_1,x_2,\ldots,x_n)$ are related to the real and imaginary parts of the sensitivities through the relationships

$$\frac{\Delta|H|}{|H|} \approx \sum_{i=1}^{n} (\mathrm{Re}\, S^{H}_{x_i}) \frac{\Delta x_i}{x_i} \tag{10}$$

and

$$\Delta \arg H \approx \sum_{i=1}^{n} (\mathrm{Im}\, S^{H}_{x_i}) \frac{\Delta x_i}{x_i} \tag{11}$$

where the sensitivities $S^{H}_{x_i}$ are defined in the usual manner, i.e.,

$$S^{H}_{x_i} = \frac{\partial H}{\partial x_i} \frac{x_i}{H} \tag{12}$$

and the Δx_i's are the deviations from their nominal values x_i. We take the x_i's to be independent random variables with mean $\bar{x}_i$ and variance σ_i^2. Thus

$$x_i = \bar{x}_i + \delta_i \tag{13}$$

with

$$E\{\delta_i\} = 0, \quad E\{\delta_i^2\} = \sigma_i^2 \tag{14}$$

Equations 10 and 11 may be written as*

$$\frac{\Delta|H|}{|H|} \approx \sum_{i=1}^{n} (\mathrm{Re}\, S_{x_i}^{H}) \frac{\delta_i}{x_i} \tag{15}$$

and

$$\Delta \arg H \approx \sum_{i=1}^{n} (\mathrm{Im}\, S_{x_i}^{H}) \frac{\delta_i}{x_i} \tag{16}$$

with $|H| = |H(\omega, x_1, x_2, \ldots x_n)|$. In view of Eqs. 13 and 14, we have

$$E\left\{\frac{\Delta|H|}{|H|}\right\} = 0 \tag{17}$$

$$E\{\Delta \arg H\} = 0 \tag{18}$$

$$E\left\{\left[\frac{\Delta|H|}{|H|}\right]^2\right\} = \sum_{i=1}^{n} (\mathrm{Re}\, S_{x_i}^{H})^2 \sigma_i^2 \tag{19}$$

and

$$E\left\{\left[\Delta \arg H\right]^2\right\} = \sum_{i=1}^{n} (\mathrm{Im}\, S_{x_i}^{H})^2 \sigma_i^2 \tag{20}$$

If the range of δ_i is restricted to $\pm \Delta_i$, then it is clear that the extrema of $\Delta|H|/|H|$ are given by

$$\max \frac{\Delta|H|}{|H|} = \sum_{i=1}^{n} |\mathrm{Re}\, S_{x_i}^{H}| \frac{\Delta_i}{x_i} \tag{21}$$

and

$$\min \frac{\Delta|H|}{|H|} = -\sum_{i=1}^{n} |\mathrm{Re}\, S_{x_i}^{H}| \frac{\Delta_i}{x_i} \tag{22}$$

One may take $\pm k\sigma_i$, with a proper choice of k, as the range of the variation of parameter[2] x_i. Then the range of variation using the extrema is

$$\mathrm{Range}_I = 2k \sum_{i=1}^{n} |\mathrm{Re}\, S_{x_i}^{H}| \frac{\sigma_i}{x_i} \tag{23}$$

---

[2] That is, if k is taken as 3, then the element will be within this range with probability better than 0.997.

*In the discussion to follow, we replace ≈ by = rather than discuss the precise meaning of the approximation.

and the range using ± $k\sigma_{\frac{\Delta|H|}{|H|}}$, where $\sigma^2_{\frac{\Delta|H|}{|H|}}$ is the variance of $\Delta|H|/|H|$, and is

$$\text{Range}_{II} = 2k\sqrt{\sum_{i=1}^{n} (\text{Re}\, S^{H}_{x_i})^2 \left(\frac{\sigma_i}{x_i}\right)^2}$$

That $\text{Range}_I \geq \text{Range}_{II}$ follows from an elementary inequality. In fact, for most cases, one will find Range_{II} to be considerably smaller than Range_I.

The third-order Chebyshev filter is used again as an illustration. The sum of the absolute value Re $S^H_{x_i}$ is calculated to be 1.3390 at $\omega/2\pi = 0.1$, 1.4033 at $\omega/2\pi = 0.15$, and 2.8755 at $\omega/2\pi = 0.2$, where as the square root of the sum of the squares of the real parts of the individual sensitivities are 0.7756, 0.7310, and 1.6730 at these frequencies. That is, for this example, the extreme range of variation Range_I is almost twice as large as a meaningful measure should be. Figure 6 shows the ranges as determined by the extrema and as determined by the variances of $\Delta|H|/|H|$ at $\omega = 1.0$ and $\omega = 1.5$, using $k = 3$. Also sketched in the figure are the probability density functions.

When the distributions of the elements are not gaussian, one can not draw the definitive result as given above. But if the number of terms in Eq. 15 is not too small and if no term in the sum clearly dominates, then the final distribution of $\Delta|H|/|H|$ will still be approximately gaussian [11]. Suppose the elements in the example of the Chebyshev filter are uniformly distributed, then the probability density of $\Delta|H|/|H|$ is that of the successive convolution of five rectangular shapes of proper width.

The result is also sketched in Fig. 6. If one takes ±3σ as the range of variation, it is seen that with probability better than .92, $\Delta|H|/|H|$ will be within the range when the elements are uniformly distributed.

ACKNOWLEDGEMENT

This research is sponsored by the Air Force Office of Scientific Research, Office of Aerospace Research, United States Air Force under AFOSR Grant 1333-67 and by the National Science Foundation under Grant GK-1439.

REFERENCES

1. Blostein, M.L., "On the Effect of Loss in Filter Networks," Proc. 3rd Allerton Conference on Circuit and System Theory, 1965; pp. 421-438.

2. Bihovski, M.L., "The Accuracy of Electric Network Analyzers" IZV an SSSR, Otd. Tekhn. Nauk, No. 8, 1948.

3. Cruz, J.B., Jr. and W. R. Perkins, "The Role of Sensitivity in the Design of Multivariable Linear Systems," Proc. NEC, 1964; pp. 742-745.

4. Cruz, J.B., Jr. and W. R. Perkins, "A New Approach to the Sensitivity Problem in Multivariable Feedback System Design," IEEE Trans. on Automatic Control, vol. AC-9, pp. 216-223; July 1964.

5. Goldstein, A.J. and F. F. Kuo, "Multiparameter Sensitivity" IRE Trans. on Circuit Theory, vol. CT-8, No. 2, pp. 177-178; June 1961.

6. Horowitz, I.M., Synthesis of Feedback Systems, Academic Press, New York, 1963.

7. Lee, S.C., "On Multiparameter Sensitivity," Third Allerton Conf. on Circuit and System Theory, October 1965, pp. 407-420.

8. Kelly, Gerry L., "Nonlinear Programming and Multiparameter Sensitivity in Linear Time-Invariant Networks," Report R-301, Coordinated Science Laboratory, University of Illinois, June 1966.

9. Hakimi, S.L. and J. B. Cruz, Jr., "Measures of Sensitivity for Linear Systems with Large Multiple Parameter Variation" IRE WESCON Convention Record, Part 2, pp. 109-115; August 1960.

10. Abramowitz, M. and I. A. Stegun, Handbook of Mathematical Functions, Dover, New York, 1965, pp. 297.

11. Papoulis, Athanasios, Probability, Random Variables, and Stochastic Processes, McGraw-Hill Book Co., 1965.

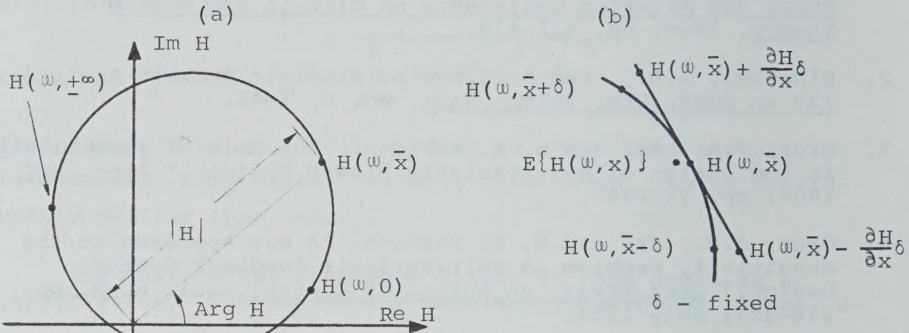

(a)
Im H
$H(\omega,\pm\infty)$
$H(\omega,\bar{x})$
$|H|$
$H(\omega,0)$
Arg H
Re H
(b)
$H(\omega,\bar{x})+\frac{\partial H}{\partial x}\delta$
$H(\omega,\bar{x}+\delta)$
$E\{H(\omega,x)\}$
$H(\omega,\bar{x})$
$H(\omega,\bar{x}-\delta)$
$H(\omega,\bar{x})-\frac{\partial H}{\partial x}\delta$
δ - fixed

Fig. 1

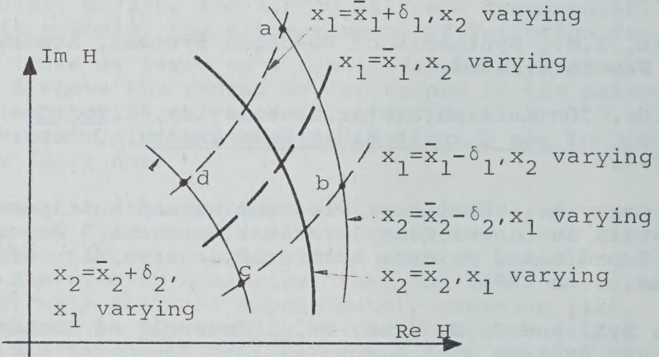

$x_1=\bar{x}_1+\delta_1, x_2$ varying
a
Im H
$x_1=\bar{x}_1, x_2$ varying
$x_1=\bar{x}_1-\delta_1, x_2$ varying
d
b
$x_2=\bar{x}_2-\delta_2, x_1$ varying
c
$x_2=\bar{x}_2+\delta_2,$
x_1 varying
$x_2=\bar{x}_2, x_1$ varying
Re H

Fig. 2

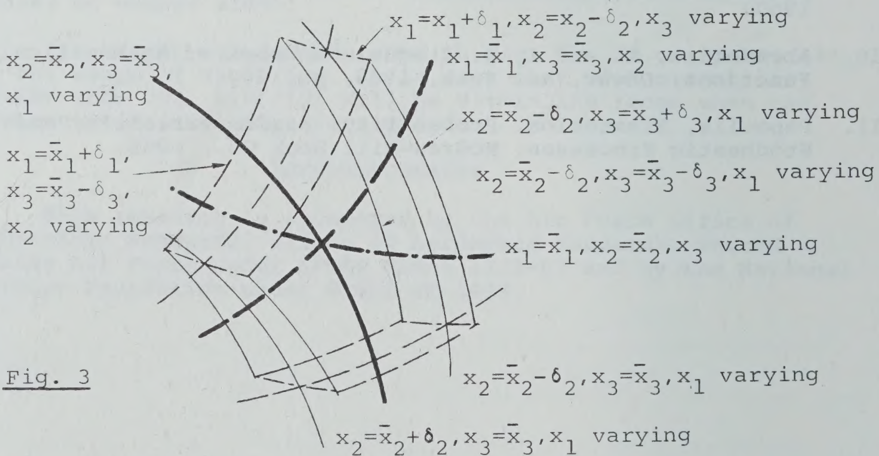

$x_1=x_1+\delta_1, x_2=x_2-\delta_2, x_3$ varying
$x_2=\bar{x}_2, x_3=\bar{x}_3,$
x_1 varying
$x_1=\bar{x}_1, x_3=\bar{x}_3, x_2$ varying
$x_2=\bar{x}_2-\delta_2, x_3=\bar{x}_3+\delta_3, x_1$ varying
$x_1=\bar{x}_1+\delta_1,$
$x_3=\bar{x}_3-\delta_3,$
x_2 varying
$x_2=\bar{x}_2-\delta_2, x_3=\bar{x}_3-\delta_3, x_1$ varying
$x_1=\bar{x}_1, x_2=\bar{x}_2, x_3$ varying
$x_2=\bar{x}_2-\delta_2, x_3=\bar{x}_3, x_1$ varying
$x_2=\bar{x}_2+\delta_2, x_3=\bar{x}_3, x_1$ varying

Fig. 3

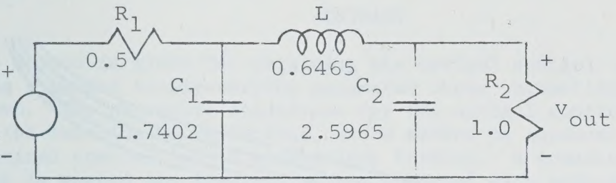

R1
0.5
L
0.6465
C1
1.7402
C2
2.5965
R2
1.0
+
v_in
-
+
v_out
-

Fig. 4

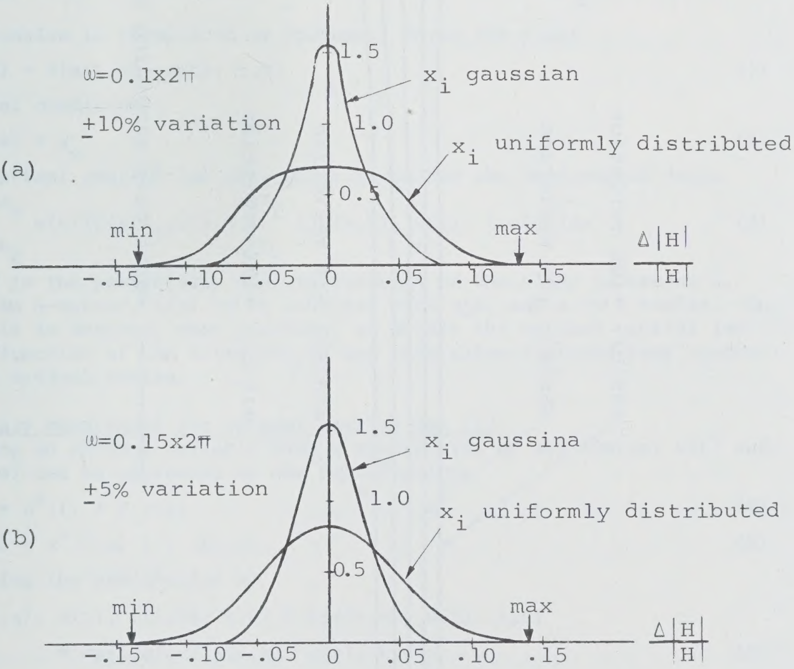

ω=0.1x2π
±10% variation
(a)
1.5
1.0
0.5
x_i gaussian
x_i uniformly distributed
min
max
-.15
-.10
-.05
0
.05
.10
.15
Δ|H| / |H|
ω=0.15x2π
±5% variation
(b)
1.5
1.0
0.5
x_i gaussina
x_i uniformly distributed
min
max
-.15
-.10
-.05
0
.05
.10
.15
Δ|H| / |H|

Fig. 6

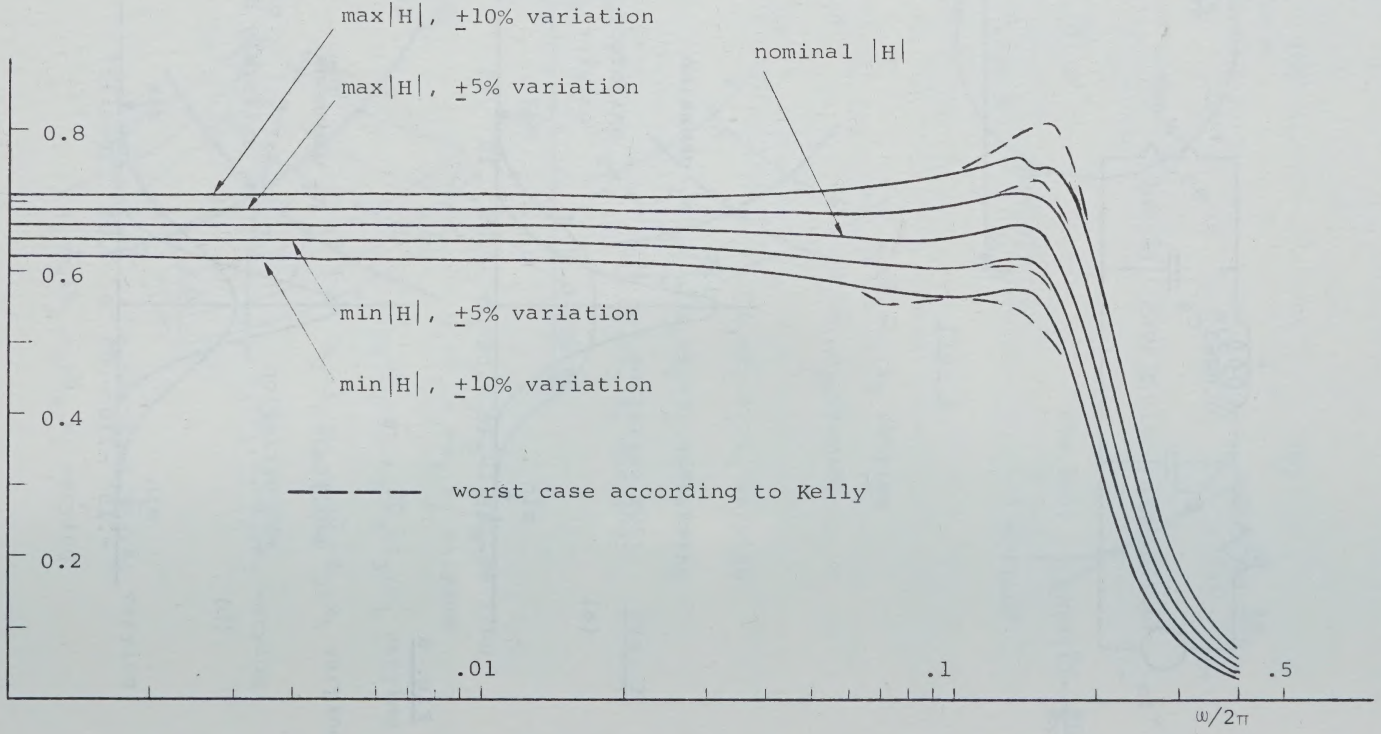

max|H|, ±10% variation
max|H|, ±5% variation
nominal |H|
min|H|, ±5% variation
min|H|, ±10% variation
worst case according to Kelly
0.8
0.6
0.4
0.2
.01
.1
.5
ω/2π

Fig. 5

OPTIMAL CONTROL OF PLANTS WITH RANDOM SLOWLY-VARYING PARAMETERS

L. BENJAMIN PATRICK and HENRY D'ANGELO
Department of Electrical Engineering, University of Denver, Colorado

ABSTRACT

A method is given for obtaining the optimal control law for plants containing a random slowly-varying parameter whose probability density function is known. The necessary conditions for the optimal control law are derived using the calculus of variations, and a method is advanced for approximating the optimal control law in closed-loop fashion. A numerical illustrative example is worked and the results are compared with both an optimal system designed about the nominal value of the parameter, and an adaptive system whose control law changes to the optimal for the existing parameter value.

MAIN RESULTS

1. Introduction

Many methods have been proposed for reducing optimal system sensitivity to small parameter variations in which the resultant design is a function of the nominal parameter value. Relatively little has been done in utilizing a-priori knowledge that may be available of the parameter's probability density function. Assuming a-priori knowledge of the parameter's probability density function, and that the random parameter is slowly time-varying, an optimal control law, which considers the total parameter variation, is derived. The resulting optimal design thus takes into account both small and large parameter variations.

The problem is formulated as follows: Given the plant

$$\dot{x}(t,\alpha) = f[x(t,\alpha), u(t); t,\alpha] \tag{1}$$

with initial conditions

$$x(t_o,\alpha) = x_o \tag{2}$$

find the optimal control law u(t) which minimizes the performance index

$$J = \int_{\alpha_\ell}^{\alpha_h} \omega(\alpha)\left\{K[x(t_f,\alpha)] + \int_{t_o}^{t_f} L[x(t,\alpha), u(t); t,\alpha]dt\right\}d\alpha \tag{3}$$

where $\omega(\alpha)$ is the probability density function of the plant parameter α, $x(t,\alpha)$ is an n-vector, $u(t)$ is an m-vector with $m \leq n$, and α is a scalar. In addition, it is desired, when possible, to obtain the optimal control law $u(t)$ as a function of the state $x(t,\alpha)$ and thus allow a closed-loop synthesis of the optimal system.

2. Necessary conditions for optimal control law [1]

Denoting an optimal variable with a superscript o, any control u(t) and state x(t,α) can be expressed by the perturbations

$$u(t) = u^o(t) + \varepsilon\, \eta(t) \tag{4}$$

$$x(t,\alpha) = x^o(t,\alpha) + \varepsilon\, \phi(t,\alpha) \tag{5}$$

Defining the Hamiltonian by

$$H[x(t,\alpha), u(t), p(t,\alpha); t,\alpha] \equiv L[x(t,\alpha), u(t); t,\alpha] + \langle p(t,\alpha), f[x(t,\alpha), u(t); t,\alpha]\rangle \tag{6}$$

it is determined that

$$J(u) - J(u^o) = \int_{\alpha_\ell}^{\alpha_h} \omega(\alpha)\{K[x(t_f,\alpha)] - K[x^o(t_f,\alpha)]\}d\alpha$$

$$+ \int_{t_o}^{t_f}\int_{\alpha_\ell}^{\alpha_h} \omega(\alpha)\{H[x(t,\alpha),\ u(t),\ p(t,\alpha);\ t,\alpha]$$

$$- H[x^o(t,\alpha),\ u^o(t),\ p^o(t,\alpha);\ t,\alpha]\}d\alpha dt$$

$$+ \int_{t_o}^{t_f}\int_{\alpha_\ell}^{\alpha_h} \omega(\alpha)\ <p(t,\alpha),\ [\dot{x}^o(t,\alpha) - x(t,\alpha)]>\ d\alpha dt \quad (7)$$

where $<\cdot, \cdot>$ denotes the scalar product. Letting ε approach zero results in

$$\int_{\alpha_\ell}^{\alpha_h} \omega(\alpha) < \frac{\partial K}{\partial x}\Big|_{o,p} - p(t_f,\alpha),\ \phi(t_f,\alpha) > d\alpha + \int_{t_o}^{t_f}\int_{\alpha_\ell}^{\alpha_h} \omega(\alpha)\ < [\frac{\partial H}{\partial x}\Big|_{o,p}$$

$$+ \dot{p}(t,\alpha)],\ \phi(t,\alpha) > d\alpha dt + \int_{t_o}^{t_f}\int_{\alpha_\ell}^{\alpha_h} \omega(\alpha)\ < \frac{\partial H}{\partial u}\Big|_{o,p},\ \eta(t) > d\alpha dt \to 0 \quad (8)$$

from which the following necessary conditions are deduced:

It is necessary that the differential equations

$$\dot{x}^o(t,\alpha) = \frac{\partial H}{\partial p} \quad (9)$$

$$\dot{p}^o(t,\alpha) = \frac{-\partial H}{\partial x} \quad (10)$$

with the boundary conditions

$$x^o(t_o,\alpha) = x_o \quad (11)$$

$$p^o(t_f,\alpha) = \frac{\partial K[x(t_f,\alpha)]}{\partial x} \quad (12)$$

and the integral equation

$$\int_{\alpha_\ell}^{\alpha_h} \omega(\xi)\ \frac{\partial H}{\partial u}\ d\xi = 0 \quad (13)$$

be satisfied.

3. The linear regulator problem

The previous results are applied to the state regulator problem in which the plant is given by

$$\dot{x}(t,\alpha) = A(\alpha)x(t,\alpha) + B(\alpha)u(t) \quad (14)$$

$$x(t_o,\alpha) = x_o \quad (15)$$

and the performance index to be minimized is

$$J = \int_{\alpha_\ell}^{\alpha_h} \omega(\alpha)\{x^*(t_f,\alpha)F\ x(t_f,\alpha) + \int_{t_o}^{t_f} [x^*(t,\alpha)Q\ x(t,\alpha)$$

$$+ u^*(t)R\ u(t)]dt\}d\alpha \quad (16)$$

(* denotes matrix transposition). The necessary conditions of equations (9) through (13) result in the following state and costate equations

$$\dot{x}(t,\alpha) = A(\alpha)\ x(t,\alpha) + B(\alpha)\ u(t) \quad (17)$$

$$\dot{p}(t,\alpha) = -Q\, x(t,\alpha) - A^*(\alpha)\, p(t,\alpha) \tag{18}$$

with boundary conditions

$$x(t_o,\alpha) = x_o \tag{19}$$

$$p(t_f,\alpha) = F\, x(t_f,\alpha) \tag{20}$$

The optimal control law $u^o(t)$, derived from equation (13), is

$$u^o(t) = -R^{-1} \int_{\alpha_\ell}^{\alpha_h} \omega(\xi) B^*(\xi) p(t,\xi)\, d\xi \tag{21}$$

4. Closed-loop optimal control

A method is now advanced which allows the control law to be approximated as a function of the state and thus provide closed-loop control. Expanding the costate $p(t,\xi)$ in a Taylor series [2] about α_o results in

$$p(t,\xi) = \sum_{i=0}^{\infty} p_i(t,\alpha_o) \frac{(\xi - \alpha_o)^i}{i!} \tag{22}$$

where

$$p_i(t,\alpha_o) \equiv \left. \frac{\partial^i p(t,\xi)}{\partial \xi^i} \right|_{\xi=\alpha_o} \tag{23}$$

Thus substituting from equation (22) in equation (21) gives

$$u(t) = -R^{-1} \int_{\alpha_\ell}^{\alpha_h} \omega(\xi) B^*(\xi) \Big[\sum_{i=0}^{\infty} p_i(t,\alpha_o) \frac{(\xi - \alpha_o)^i}{i!} \Big] d\xi \tag{24}$$

For notational simplicity c_i is defined as

$$c_i \equiv \int_{\alpha_\ell}^{\alpha_h} \omega(\xi) B^*(\xi) \frac{(\xi - \alpha_o)^i}{i!}\, d\xi \tag{25}$$

Thus

$$u(t) = -R^{-1} \sum_{i=0}^{\infty} c_i p_i(t,\alpha_o)$$

$$= -R^{-1} \hat{c}^* \hat{p}(t,\alpha_o) \tag{26}$$

where $\hat{c}$ and $\hat{p}(t,\alpha_o)$ are the appropriate infinite vectors. Differentiating equation (18) with respect to α gives

$$\dot{p}_i(t,\alpha_o) = -Q\, v_i(t,\alpha_o) - \left. \frac{\partial^i [A^*(\xi) p(t,\xi)]}{\partial \xi^i} \right|_{\xi=\alpha_o} \tag{27}$$

where $v_i(t,\alpha_o)$ denotes the i^{th} trajectory sensitivity [3] defined by

$$v_i(t,\alpha_o) \equiv \left. \frac{\partial^i x(t,\xi)}{\partial \xi^i} \right|_{\xi=\alpha_o}$$

Hence, the following matrix equation can be formed:

$$\begin{vmatrix} \dot{p}(t,\alpha_o) \\ \dot{p}_1(t,\alpha_o) \\ \dot{p}_2(t,\alpha_o) \\ \vdots \\ \dot{p}_r(t,\alpha_o) \\ \vdots \end{vmatrix} = - \begin{vmatrix} A^*(\alpha_o) & 0 & 0 & \dots & 0 & \dots \\ A_1^*(\alpha_o) & A^*(\alpha_o) & 0 & \dots & 0 & \dots \\ A_2^*(\alpha_o) & 2A_1^*(\alpha_o) & A^*(\alpha_o) & \dots & 0 & \dots \\ \vdots & \vdots & \vdots & & \vdots & \\ A_r^*(\alpha_o) & rA_{r-1}^*(\alpha_o) & \sum_{i=1}^{r-1} iA_{r-2}^*(\alpha_o) & \dots & A^*(\alpha_o) & \dots \\ \vdots & & \vdots & & \vdots & \end{vmatrix} \begin{vmatrix} p(t,\alpha_o) \\ p_1(t,\alpha_o) \\ p_2(t,\alpha_o) \\ \vdots \\ p_r(t,\alpha_o) \\ \vdots \end{vmatrix}$$

$$- \begin{vmatrix} Q & 0 & 0 & \dots & 0 & \dots \\ 0 & Q & 0 & \dots & 0 & \dots \\ 0 & 0 & Q & \dots & 0 & \dots \\ \vdots & \vdots & \vdots & & \vdots & \\ 0 & 0 & 0 & \dots & Q & \dots \\ \vdots & \vdots & \vdots & & \vdots & \end{vmatrix} \begin{vmatrix} x(t,\alpha_o) \\ v_1(t,\alpha_o) \\ v_2(t,\alpha_o) \\ \vdots \\ v_r(t,\alpha_o) \\ \vdots \end{vmatrix} \tag{29}$$

where

$$A_i(\alpha_o) = \left. \frac{\partial^i A(\alpha)}{\partial \alpha^i} \right|_{\alpha=\alpha_o} \tag{30}$$

with boundary conditions derived from equation (20):

$$\begin{vmatrix} p(t_f,\alpha_o) \\ p_1(t_f,\alpha_o) \\ p_2(t_f,\alpha_o) \\ \vdots \\ p_r(t_f,\alpha_o) \\ \vdots \end{vmatrix} = \begin{vmatrix} F & 0 & 0 & \dots & 0 & \dots \\ 0 & F & 0 & \dots & 0 & \dots \\ 0 & 0 & F & \dots & 0 & \dots \\ \vdots & \vdots & \vdots & & \vdots & \\ 0 & 0 & 0 & \dots & F & \dots \\ \vdots & \vdots & \vdots & & \vdots & \end{vmatrix} \begin{vmatrix} x(t_f,\alpha_o) \\ v_1(t_f,\alpha_o) \\ v_2(t_f,\alpha_o) \\ \vdots \\ v_r(t_f,\alpha_o) \\ \vdots \end{vmatrix} \tag{31}$$

Using matrix notation equations (29) and (31) are rewritten:

$$\dot{\hat{p}}(t,\alpha_o) = \tilde{A}(\alpha_o)\hat{p}(t,\alpha_o) - \hat{Q}\hat{x}(t,\alpha_o) \tag{32}$$

$$\hat{p}(t_f,\alpha_o) = \hat{F}\,\hat{x}(t_f,\alpha_o) \tag{33}$$

where $\tilde{A}(\alpha_o)$, $\hat{Q}$, $\hat{F}$ and $\hat{x}(t,\alpha_o)$ are appropriately defined. Similarly, differentiating equation (17) with respect to α results in

$$\begin{vmatrix} \dot{x}(t,\alpha_o) \\ \dot{v}_1(t,\alpha_o) \\ \dot{v}_2(t,\alpha_o) \\ \vdots \\ \dot{v}_r(t,\alpha_o) \\ \vdots \end{vmatrix} = \begin{vmatrix} A(\alpha_o) & 0 & 0 & \dots & 0 & \dots \\ A_1(\alpha_o) & A(\alpha_o) & 0 & \dots & 0 & \dots \\ A_2(\alpha_o) & 2A_1(\alpha_o) & A(\alpha_o) & \dots & 0 & \dots \\ \vdots & \vdots & \vdots & & \vdots & \\ A_r(\alpha_o) & rA_{r-1}(\alpha_o) & \sum_{i=1}^{r-1} iA_{r-2}(\alpha_o) & \dots & A(\alpha_o) & \dots \\ \vdots & \vdots & \vdots & & \vdots & \end{vmatrix} \begin{vmatrix} x(t,\alpha_o) \\ v_1(t,\alpha_o) \\ v_2(t,\alpha_o) \\ \vdots \\ v_r(t,\alpha_o) \\ \vdots \end{vmatrix}$$

$$+ \left| \begin{matrix} B(\alpha_o) \\ B_1(\alpha_o) \\ B_2(\alpha_o) \\ \vdots \\ B_r(\alpha_o) \\ \vdots \end{matrix} \right| u(t) \tag{34}$$

where

$$B_i(\alpha_o) = \left. \frac{\partial^i B(\alpha)}{\partial \alpha} \right|_{\alpha=\alpha_o} \tag{35}$$

with boundary conditions derived from equation (19):

$$\left| \begin{matrix} x(t_o,\alpha_o) \\ v_1(t_o,\alpha_o) \\ v_2(t_o,\alpha_o) \\ \vdots \\ v_r(t_o,\alpha_o) \\ \vdots \end{matrix} \right| = \left| \begin{matrix} x_o \\ 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \end{matrix} \right| \tag{36}$$

Using matrix notation equations (34) and (36) are rewritten:

$$\dot{\hat{x}}(t,\alpha_o) = \hat{A}(\alpha_o)\hat{x}(t,\alpha_o) + \hat{B}(\alpha_o)u(t) \tag{37}$$

$$\hat{x}(t_o,\alpha_o) = \hat{x}_o$$

where $\hat{A}(\alpha_o)$, $\hat{B}(\alpha_o)$ and $\hat{x}_o$ are appropriately defined. Substituting from equation (26) in equation (37) gives

$$\dot{\hat{x}}(t,\alpha_o) = \hat{A}(\alpha_o)\hat{x}(t,\alpha_o) - \hat{B}(\alpha_o)R^{-1}\hat{c}^*p(t,\alpha_o) \tag{39}$$

Thus equations (32) and (39) provide simultaneous equations for $x(t,\alpha_o)$ and $p(t,\alpha_o)$:

$$\dot{\hat{x}}(t,\alpha_o) = \hat{A}(\alpha_o)\hat{x}(t,\alpha_o) - \hat{B}(\alpha_o)R^{-1}\hat{c}^*p(t,\alpha_o) \tag{40a}$$

$$\dot{\hat{p}}(t,\alpha_o) = - \hat{Q}\,\hat{x}(t,\alpha_o) - \tilde{A}(\alpha_o)\hat{p}(t,\alpha_o) \tag{40b}$$

with the boundary conditions

$$\hat{x}(t_o,\alpha_o) = \hat{x}_o \tag{41a}$$

$$\hat{p}(t_f,\alpha_o) = \hat{F}\,\hat{x}(t_f,\alpha_o) \tag{41b}$$

Equations (40) and (41) can be utilized [4, 5] to show that

$$\hat{p}(\alpha_o) = \hat{K}(t,\alpha_o)\hat{x}(t,\alpha_o) \tag{42}$$

where $\hat{K}(t,\alpha_o)$ is the solution to a matrix Riccati equation:

$$\dot{\hat{K}}(t,\alpha_o) + \hat{K}(t,\alpha_o)\hat{A}(\alpha_o) + \tilde{A}(\alpha_o)\hat{K}(t,\alpha_o)$$

$$- \hat{K}(t,\alpha_o)\hat{B}(\alpha_o)R^{-1}\hat{c}^*\hat{K}(t,\alpha_o) + \hat{Q} = 0 \tag{43}$$

Substitution of equation (42) in equation (26) provides a relationship for $u(t)$ in terms of $\hat{x}(t,\alpha_o)$, i.e., in terms of the state $x(t,\alpha_o)$ and all the trajectory sensitivity vectors $v_i(t,\alpha_o)$ $i = 1, 2, \ldots$ In particular

$$u(t) = - R^{-1}\hat{c}^{*}\hat{K}(t,\alpha_o)\hat{x}(t,\alpha_o) \tag{44}$$

The trajectory sensitivity vectors are easily generated [6].

A simulation of such a closed-loop optimal control system is shown in Figure 1. It should be noted that in generating the trajectory sensitivity vectors $v_i(t,\alpha_o)$ knowledge of the plant parameter α is implied. Thus if gains are not changed as the plant parameter α changes, the control law generated in this fashion will be only an approximation to the optimal control law. Experience shows that this is generally a good approximation. Further, in any practical situation only a finite number of trajectory sensitivity vectors will be generated. Thus all the infinite-order equations are approximated by finite-order equations.

5. Illustrative numerical example

Consider the first-order system characterized by

$$\dot{x} = - \alpha x + u \tag{45}$$

Knowing that the parameter α is uniformly distributed on the interval (0,2), it is desired to obtain the control u that minimizes the performance index

$$J = \int_0^2 \frac{1}{2}\int_0^\infty \frac{1}{2}(x^2 + u^2)dt\, d\alpha \tag{46}$$

Since this is an infinite time regulator problem, the matrix Riccati differential equation reduces to an algebraic equation. Truncating the Taylor expansion of $p(t,\xi)$ given in equation (22) after 3 terms results in the Riccati equation (43) being

$$\begin{vmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{vmatrix} \begin{vmatrix} -\alpha_o & 0 & 0 \\ -1 & -\alpha_o & 0 \\ 0 & -2 & -\alpha_o \end{vmatrix} + \begin{vmatrix} -\alpha_o & 0 & 0 \\ -1 & -\alpha_o & 0 \\ 0 & -2 & -\alpha_o \end{vmatrix} \begin{vmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{vmatrix}$$

$$- \begin{vmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{vmatrix} \begin{vmatrix} 1 \\ 0 \\ 0 \end{vmatrix} (1)\ [1,\ 1- \alpha_o,\ \frac{2}{3} - \alpha_o + \frac{1}{2}\alpha_o^{2}] \begin{vmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{vmatrix}$$

$$+ \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{vmatrix}$$

The approximate control, in accordance with equation (44), is

$$u(t) = (-1)\ [1,\ 1- \alpha_o,\ \frac{2}{3} - \alpha_o + \frac{1}{2}\alpha_o^{2}] \begin{vmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{vmatrix} \begin{vmatrix} x(t,\alpha_o) \\ v_1(t,\alpha_o) \\ v_2(t,\alpha_o) \end{vmatrix} \tag{48}$$

A block diagram showing the simulation of this system is shown in Figure 2.

Figure 3a shows a comparison of the cost C defined by

$$C = \int_0^\infty (x^2 + u^2)dt \tag{49}$$

over the range of α, (0,2), between a system designed to minimize J (the expected value of C), using such a 3-term approximation $\alpha_o = 1$ (Figure 3b shows similar results for a 5-term approximation), and the optimal regulator minimizing C at a nominal $\alpha = 1$. Also shown in this figure is the cost that would be obtained by an optimally adaptive system, i.e., a system with a control law u(t) that is adjusted, on line, to be optimal for the particular value of α that the plant is noted to have. Clearly, this is the minimal cost possible. It is noted that the expected value of the cost using this simple approximation to the control minimizing J is significantly decreased from that not taking into account variations in the parameter α.

Figure 4 shows that the trajectories of a system so designed are less sensitive to variations in the parameter α.

6. Acknowledgements

The authors gratefully acknowledge the support of the National Science Foundation (Grant No. GK 274) and NASA (Grant No. NsG(T)49 and Grant No. NsG-518).

REFERENCES

[1] Athans, M. and Falb, P., Optimal Control, McGraw-Hill, 1966.

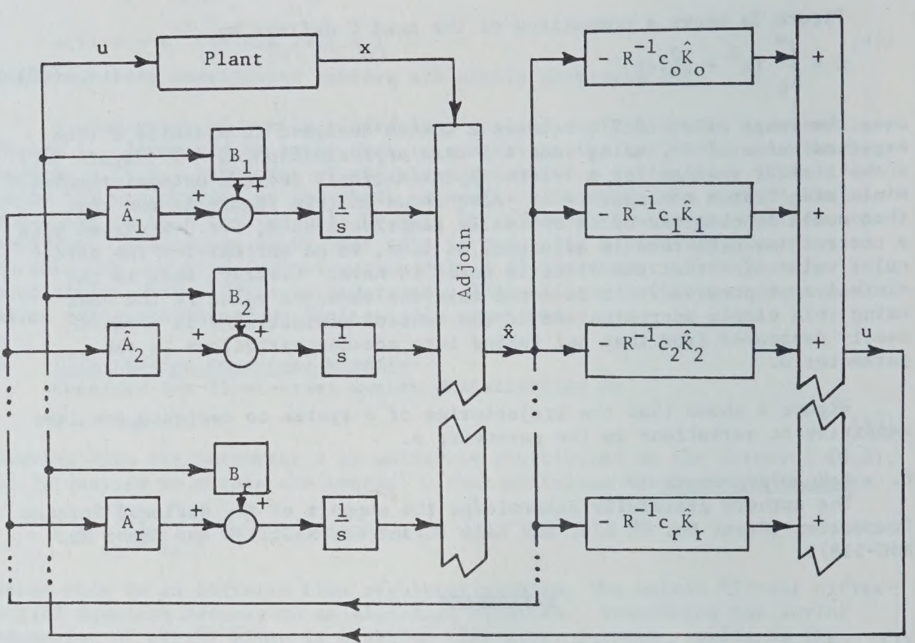
[2] D'Angelo, H., Moe, M. L., and Hendricks, T. C., "Trajectory Sensitivity of an Optimal Control System", Proceedings, Fourth Annual Allerton Conference on Circuit and System Theory, 1966.

[3] Dorato, P., "On Sensitivity in Optimal Control Systems", IEEE Transactions on Automatic Control, Vol. AC-8, No. 3, July, 1963.

[4] Kalman, R. E., "Contributions to the Theory of Optimal Control", Baletin Sociedad Matematica Mexicana, Vol. 5, 1960.

[5] Reed, W. T., "A Matrix Differential Equation of Riccati Type", American Journal of Mathematics, Vol. 68, 1946.

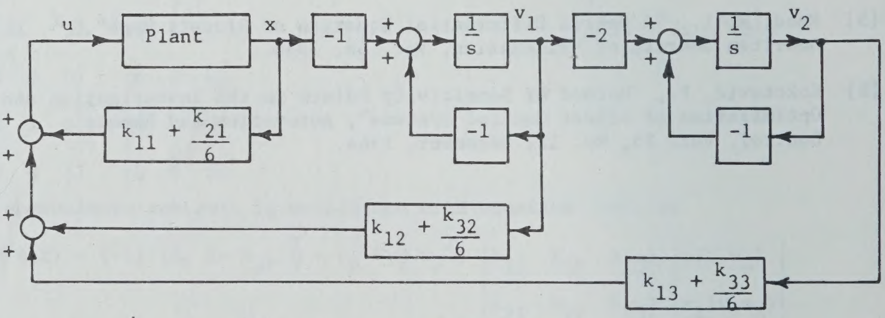
[6] Kokotović, P., "Method of Sensitivity Points in the Investigation and Optimization of Linear Control Systems", Automation and Remote Control, Vol. 25, No. 12, December, 1964.


u
Plant
x
B_1
$\hat{A}_1$
$\frac{1}{s}$
B_2
$\hat{A}_2$
B_r
$\hat{A}_r$
+
Adjoin
$\hat{x}$
$- R^{-1}c_o\hat{K}_o$
$- R^{-1}c_1\hat{K}_1$
$- R^{-1}c_2\hat{K}_2$
$- R^{-1}c_r\hat{K}_r$
u

$\hat{A}_i$ denotes the i^{th} row of $\hat{A}$, $i = 0, \ldots, \infty$

$\hat{K}_i$ denotes the i^{th} row of $\hat{K}$, $i = 0, \ldots, \infty$

Figure 1. Simulation of Optimal System Characterized by Equations 40 and 41

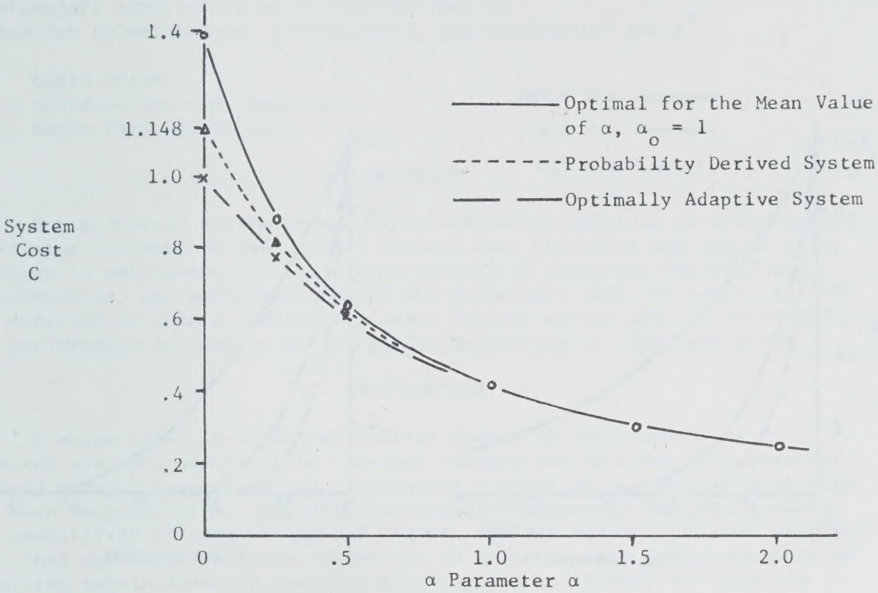

u
Plant
x
-1
+
$\frac{1}{s}$
v_1
-2
v_2
$k_{11} + \frac{k_{21}}{6}$
$k_{12} + \frac{k_{32}}{6}$
$k_{13} + \frac{k_{33}}{6}$

$k_{11} = .3877$ $k_{12} = .0342$ $k_{13} = - .0125$

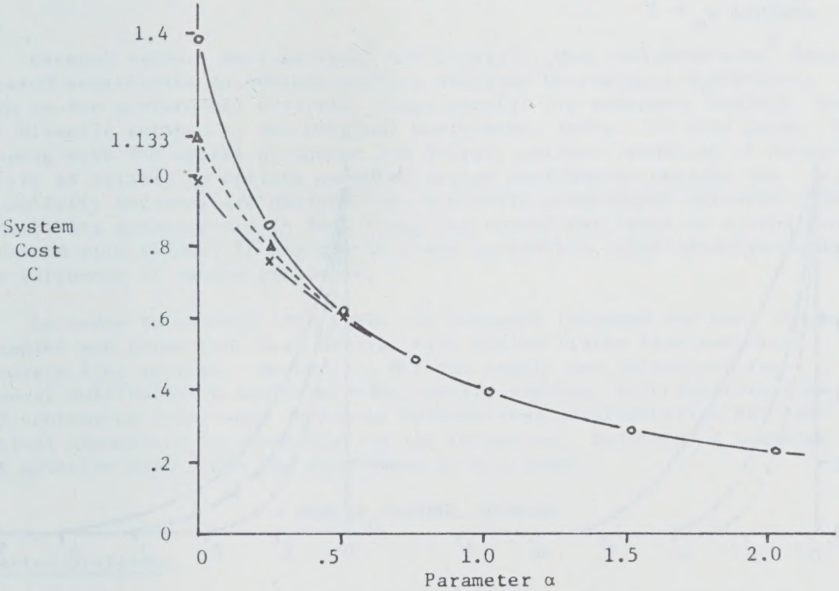
$k_{21} = - .3351$ $k_{22} = .4471$ $k_{23} = .0171$

$k_{31} = .6205$ $k_{32} = - .8753$ $k_{33} = .4629$

Figure 2. Simulation of Third-Order Approximation to Optimal System of Figure 1. for the Plant, $\dot{x} = - \alpha x + u$, $\alpha_o = 1$

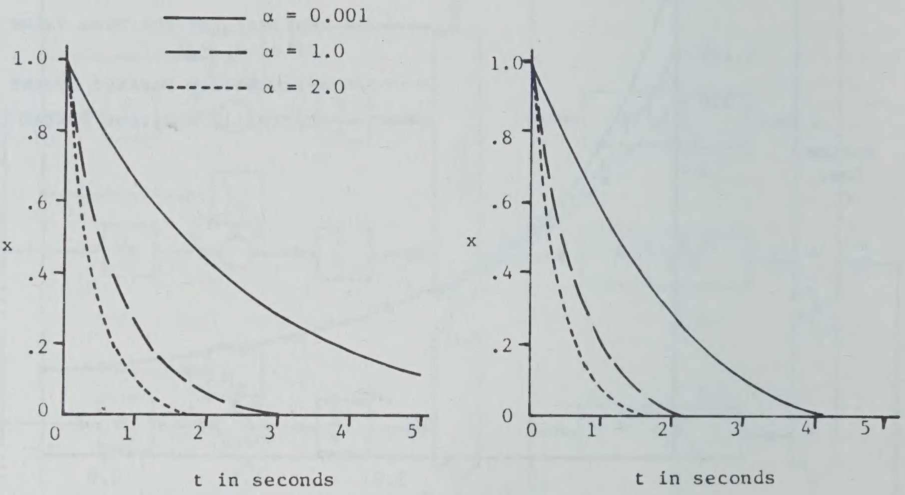


(a) 3rd-order approximation



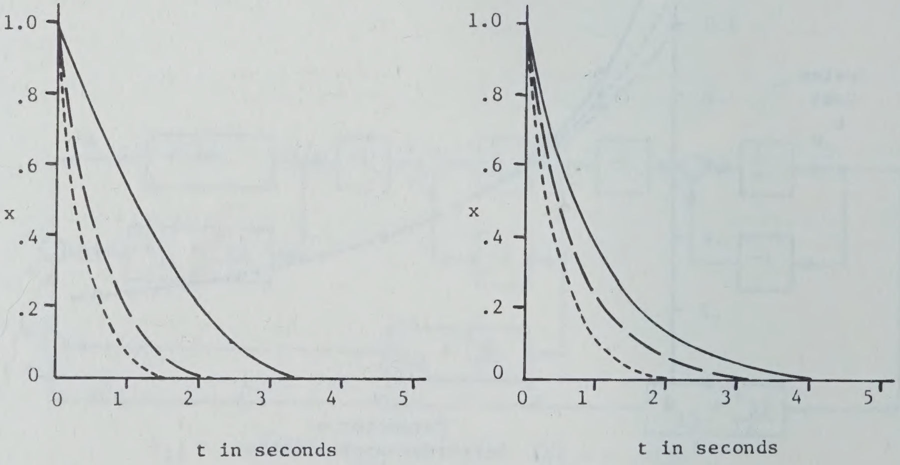
(b) 5th-order approximation

Figure 3. Comparisons of System Costs Over the Range of the Parameter α



System minimizing cost C for nominal $\alpha_o = 1$

3rd-order approximation



5th-order approximation

Optimal adaptive system

Figure 4. State Trajectories Over the Range of α

PERFORMANCE SENSITIVITY OF CONTROL SYSTEMS TO PARAMETER UNCERTAINTIES, DISTURBANCES, AND MEASUREMENT NOISE*

LEWIS MEIER
Stanford Research Institute
Menlo Park, California

ABSTRACT

For a control system in which the controller consists of a linearized estimator followed by the optimal control law, the first and second order changes in performance of the system because of parameter uncertainties, disturbances, and measurement noise are described. How the controller may be modified to reduce sensitivity in an optimal manner and the sensitivity of performance to changes in controller parameters is also considered.

INTRODUCTION

A major topic in classical control theory is the sensitivity of a control system's operation to internal changes and external disturbances; indeed a major reason for use of feedback control is to reduce sensitivity to such uncertainties. Recently interest has arisen in the relationship of sensitivity to optimal control theory. Rohrer and Sobral[1] have pointed out that one is not so much interested in insuring that the performance of a system remain fixed in the face of changing conditions, but that the system operate near optimum for the conditions present--whatever they may be.

Several authors such as Tuel[2] and D'Angelo, Moe, and Hendricks[3] have related sensitivity to optimal control problems by adding a sensitivity term to the performance criteria; their sensitivity measures, however, are not directly related to the original performance index. In this paper, in keeping with the spirit of Rohrer and Sobral, optimum reduction of sensitivity is related to optimum expected system performance because the sensitivity measures are derived from the basic performance criteria. The sensitivity measures are in fact first and second derivates of system performance with respect to changes in plant parameters, controller parameters, and variances of random processes.

In order to clearly illustrate the concepts involved two very simple examples are presented; both dealing with scalar linear time invariant discrete time systems. However, a similar result may be derived for general multivariable nonlinear time varying systems, both discrete time and continuous time; only increased mathematical sophistication not conceptual innovation is necessary for the extension. Reference 4 contains the detailed derivation for continuous time systems.

A SIMPLE CONTROL PROBLEM

Problem Statement

Consider finding the optimal control for the plant described by

$$X_{k+1} = X_k + U_k \quad ; \quad k = 0,1 \quad , \tag{1}$$

---

*The research described in this paper is based on research done for NASA Ames Research Center on Contract NAS2-3476.

where the performance index is

$$J = 2\,U_0^2 + 2\,U_1^2 + X_2^2 \quad . \tag{2}$$

Problem Solution

Substitution of Eq. (1) into Eq. (2) with the initial condition $X_0 = 4$ yields

$$J = 2\,U_0^2 + 2\,U_1^2 + (4 + U_0 + U_1)^2 \quad . \tag{3}$$

By differentiation the optimal values of U_0, U_1, and J can be found:

$$U_0^* = U_1^* = -1 \quad , \quad J^* = 8 \quad . \tag{4}$$

The optimal feedback controller must be linear; hence it may be obtained by inspection from the open loop controller given in Eq. (4):

$$U_0^* = -\frac{1}{4}\,X_0$$

$$U_1^* = -\frac{1}{3}\,X_1$$

$$J^* = \frac{1}{2}\,X_0^2 \quad . \tag{5}$$

Sensitivity of Performance to Parameter Uncertainties

Now suppose that the actual state equation of the system is

$$X_{k+1} = (1 + \Delta\alpha)\,X_k + U_k \qquad k = 0,1 \quad . \tag{6}$$

An initial state of $X_0 = 4$ will again be considered. For the modified system a modified controller of the form

$$U_0 = -1 + \Delta\,U_0 \quad , \quad U_1 = -1 + \Delta\,U_1 \tag{7}$$

will be considered.

Substitution of Eqs. (6) and (7) into Eq. (2) gives

$$\begin{aligned} J = {} & 8 + 28\,\Delta\alpha + 65\,\Delta\alpha^2 + 56\,\Delta\alpha^3 + 16\,\Delta\alpha^4 \\ & + (18 + 22\,\Delta\alpha + 8\,\Delta\alpha^2)\,\Delta\alpha\,\Delta\,U_0 + (14 + 8\,\Delta\alpha)\,\Delta\alpha\,\Delta\,U_1 \\ & + (3 + \Delta\alpha + \Delta\alpha^2)\,\Delta\,U_0^2 + 2(1 + \Delta\alpha)\,\Delta\,U_0\,\Delta\,U_1 + 3\,\Delta\,U_1^2 \end{aligned} \tag{8}$$

$$J \approx 8 + 28\ \Delta\alpha + 65\ \Delta\alpha^2 + 18\ \Delta\alpha\ \Delta\ U_0 + 14\ \Delta\alpha\ \Delta\ U_1$$
$$+ 3\ \Delta\ U_0^2 + 2\ \Delta\ U_0\ \Delta\ U_1 + 3\ \Delta\ U_1^2 \quad . \tag{9}$$

Note that because the optimal control for $\Delta\alpha = 0$ is $U_0 = U_1 = -1$, $\Delta\ U_0$ and $\Delta\ U_1$ appear only in second and higher order terms.

If $\Delta\alpha$ can be determined then $\Delta\ U_0$ and $\Delta\ U_1$ can be selected to minimize J as given in Eq. (8). Approximately optimal values may be obtained by minimization of Eq. (9); the result is

$$\Delta\ U_0^* \approx 2.5\ \Delta\alpha$$

$$\Delta\ U_1^* \approx 1.5\ \Delta\alpha$$

$$J^* \approx 8 + 28\ \Delta\alpha + 32\ \Delta\alpha^2 \quad . \tag{10}$$

Table 1 gives a comparison between the approximate and exact computations of $\Delta\ U_0$ and $\Delta\ U_1$. Two performance figures are given for the approximate computations; the upper performance is the exact performance found by substituting the approximately optimal $\Delta\ U_1$ and $\Delta\ U_2$ into Eq. (8), and the lower number is the approximate performance as given by Eq. (9). Since the approximate and exact optimum controls give almost identical performance, even though the approximate performance computation is in error, in the sequel all terms above second order will be dropped. It is to be understood that the term optimal when used hereafter means optimal to within second order.

Table 1

	$\Delta\alpha = +0.2$		$\Delta\alpha = -0.2$	
	Exact	Approximate	Exact	Approximate
ΔU_0	-0.545	-0.500	0.438	0.300
ΔU_1	-0.300	-0.300	0.300	0.300
ΔJ	6.77	6.97	-4.39	-4.37
$\sim\Delta J$		6.70		-4.30

If it is impossible to determine $\Delta\alpha$ and if open loop control is used $\Delta\ U_0 = \Delta\ U_1 = 0$ and the performance $J^{O.L.}$ is given by

$$J^{O.L.} = 8 + 28\ \Delta\alpha + 65\ \Delta\alpha^2 \quad . \tag{11}$$

If closed loop control is used $\Delta\ U_0 = 0$, but

$$\Delta\ U_1 = -\frac{1}{3}\ \Delta\alpha \quad . \tag{12}$$

Substitution of these values in Eq. (9) yields

$$J^{C.L.} = 8 + 28\ \Delta\alpha + 51.7\ \Delta\alpha^2 \quad . \tag{13}$$

Figure 1 gives graphs of J^*, $J^{O.L.}$, and $J^{C.L.}$ as given by Eqs. (9), (11), and (13). Two comments are in order: First, note that the first order terms are the same for all three equations. The fact that the first order sensitivity of optimal performance to parameter changes is the same for open and closed loop control was first noted by Pagurek.[5] From the figure it is obvious why this has to be since $J^{O.L.}$ and $J^{C.L.}$ are smooth, must lie above J^* and must coincide with J^* at $\Delta\alpha = 0$. Second, the second order sensitivity is greater for open loop than closed loop control. There are other examples, however, where the closed loop control has higher second order sensitivity to parameter changes that open loop control.

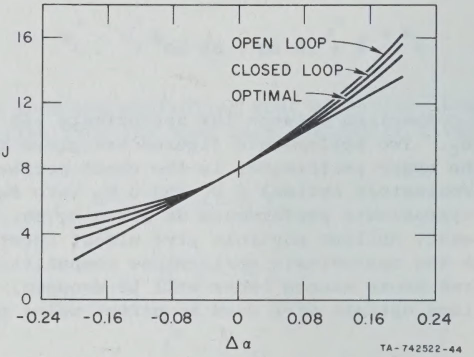


FIG. 1 COMPARISON OF PERFORMANCES

A SLIGHTLY MORE COMPLICATED CONTROL PROBLEM

Problem Statement

Now consider the plant described by

$$X_{k+1} = (1 + \Delta\alpha)\ X_k + (1 + \Delta\beta)\ U_k + W_k \qquad k = 0,\ 1 \tag{14}$$

where W_k is white noise of mean zero and variance $\hat{Q}_k$ and X_0 has mean 4 and variance $\bar{P}$. The plant is to be controlled on the basis of measurements Z_k, which are related to the plant state by

$$Z_k = X_k + V_k \quad , \quad k = 0,\ 1 \quad , \tag{15}$$

where V_k is white noise with zero mean and variance $\hat{R}_k$. Because of the presence of random variables, the performance index must be modified to

$$J = E\left[2\ U_0^2 + 2\ U_1^2 + X_2^2\right] \quad . \tag{16}$$

Form of the Controller

The controller is broken down into an estimator and a control law. The estimator is described by

$$\hat{X}_{k+1/k} = (1 + \Delta\hat{\alpha}_k)\,\hat{X}_{k/k} + (1 + \Delta\hat{B}_k)\,U_k \quad , \quad k = 0 \quad ;$$

$$\hat{X}_{k/k} = \hat{X}_{k/k-1} + \hat{K}_k\,(Z_k - \hat{X}_{k/k-1}) \quad , \quad k = 0,\ 1 \quad ;$$

$$\Delta\hat{\beta}_k = \Delta\hat{\alpha}_{k-1} + \hat{K}_k^{\alpha}\,(Z_k - \hat{X}_{k/k-1}) \quad , \quad k = 0,\ 1 \quad ;$$

$$\Delta\hat{\beta}_k = \Delta\hat{\beta}_{k-1} + \hat{K}_k^{\beta}\,(Z_k - \hat{X}_{k/k-1}) \quad , \quad k = 0,\ 1 \quad ; \tag{17}$$

where $\hat{X}_{-1} = 4$, $\Delta\hat{\alpha}_{-1} = \Delta\hat{\beta}_{-1} = 0$.

The control law is described in terms of the nominal trajectory

$$X_{k+1}^0 = X_k^0 + U_k^0 \quad ; \quad k = 0,\ 1 \tag{18}$$

where X_0^0 is taken to be 4. The control law is

$$U_k = U_k^0 - K_k\,(\hat{X}_{k/k} - X_k^0) - K_k^{\alpha}\,\Delta\hat{\alpha}_k - K_k^{\beta}\,\Delta\hat{\beta}_k \quad , \quad k = 0,\ 1 \quad . \tag{19}$$

Note that the controller form has provision for adaptive control by estimation of the parameters $\Delta\alpha$ and $\Delta\beta$ from the measurements. A controller is specified by giving the values of $\hat{K}_k$, $\hat{K}_k^{\alpha}$, $\hat{K}_k^{\beta}$, U_k^0, K_k, K_k^{α}, and K_k^{β}.

Sensitivity of Performance to Parameter Uncertainties, Disturbances, and Measurement Noise

Now consider use of a controller that consists of the optimal estimator, ignoring parameter variations, followed by the optimal deterministic closed loop control law. This controller is described by Eqs. (17), (18), and (19) with $\hat{K}_0^{\alpha} = \hat{K}_1^{\alpha} = \hat{K}_0^{\beta} = \hat{K}_1^{\beta} = K_0^{\alpha} = K_1^{\alpha} = K_0^{\beta} = K_1^{\beta} = 0$, $U_0^0 = U_1^0 = -1$, $K_0 = 1/4$, $K_1 = 1/3$ and

$$\hat{K}_0 = \frac{\bar{P}}{\bar{P} + \hat{R}_0} \quad , \quad \hat{K}_1 = \frac{\bar{P}\,\hat{R}_0 + \hat{Q}_0\,(\bar{P} + \bar{R}_0)}{\bar{P}\,\hat{R}_0 + (\bar{Q}_0 + \hat{R}_1)\,(\bar{P} + \hat{R}_0)} \quad . \tag{20}$$

With considerable patience Eqs. (14), (15), (17), (18), (19), and (20) can be substituted into Eq. (16) to yield

$$\begin{aligned} J = {} & 8 + \frac{1}{2}\,\bar{P} + \frac{2}{3}\,\hat{Q}_1 + \hat{Q}_2 + \frac{1}{3}\,\hat{P}_0 + \frac{1}{6}\,\hat{P}_1 \\ & + 28\,\Delta\alpha - 8\,\Delta\beta + 32\,\Delta\alpha + 16\,\Delta\alpha\,\Delta\beta + 4\,\Delta\beta^2 \\ & + (33 - \frac{56}{3}\lambda\hat{K} + \frac{16}{3}\,\hat{K}^2)\,\Delta\alpha^2 \end{aligned} \tag{21}$$

where $\hat{P}_0$ and $\hat{P}_1$ are the errors in estimation of X_k at times 0 and 1 respectively.

Consider Eq. (21) term by term. The first term is just the deterministic cost, the second term is the cost due to a priori uncertainty about initial conditions, the third and fourth terms the costs due to disturbances, the fifth and sixth terms the costs due to estimation error caused by the presence of measurement noise, the seventh and eighth terms the first order costs due to parameter uncertainties and the remaining term the second order cost due to parameter uncertainty.

The coefficients of the variances in terms two through eight may be interpreted as the sensitivities to initial condition uncertainties, disturbances, and estimation errors. The two first order parameter sensitivities, which are dependent only on the nominal controls U_0^0 and U_0^1, give the sensitivity of the performance to the value of the parameter. They tell which parameter should be changed in what direction in order to improve performance. In this example a change $\Delta\alpha$ in the negative direction is over 3 times as effective as a similar change $\Delta\beta$ in the positive direction.

The first three second order sensitivities to parameter variations, which depend only upon the nominal controls U_0^0 and U_0^1, and the control gains K_0 and K_1, indicate the maximum improvement that can be obtained by determining the actual value of $\Delta\alpha$ and $\Delta\beta$ and modifying the control law accordingly. The final second order term indicates the additional cost incurred from not knowing $\Delta\alpha$. Note that this sensitivity depends upon the estimation gain. With $\hat{R}_1 = \infty$, i.e., open loop control, this term is 33 and with $\hat{R}_1 = 0$, i.e., perfect closed loop control, this term is 29.7--results that agree with those given in Eqs. (11) and (13). In general there would be similar terms involving $\Delta\beta$, but in this example there is no cost incurred by lack of knowledge of $\Delta\beta$.

Optimization

In this section the problem of reducing the sensitivity to parameter variations is considered. It is known from linear control theory that values of the controller parameters used above yields the minimum value of J exclusive of parameter uncertainties; hence any reduction of parameter uncertainties is at the expense of increased cost from these terms. To perform an optimal trade-off statistics of the parameter uncertainties must be provided.

For simplicity it will be assumed that $\overline{P} = \hat{Q}_k = \hat{R}_k = \Delta\beta = 0$; therefore the problem is one of trading-off deterministic cost with sensitivity to uncertainty in $\Delta\alpha$. It will further be assumed that $E(\Delta\alpha) = 0$ and $E(\Delta\alpha)^2 = 0.06$. Since the measurements are perfect it is obvious that $\hat{K}_0$ and $\hat{K}_1$ should be set to zero; furthermore, the first measurement provides no information about $\Delta\alpha$, but with the second measurement $\Delta\alpha$ is known exactly--hence $\hat{K}_0^\alpha = 0$ and $\hat{K}_1^\alpha = 1$. The controller parameters to be determined are U_0^0, U_1^0, K_1, and K_1^α (since $X_0 = X_0^0$ and $\Delta\hat{\alpha}_0 = 0$, K_0 and K_0^α are irrelevent). In terms of these parameters the performance is given by

$$J = 2\left(U_0^0\right)^2 + 2\left(U_1^0\right)^2 + \left(4 + U_0^0 + U_1^0\right)^2$$

$$+ 0.06\left[2\left(4\ K_1 + K_1^\alpha\right)^2 + \left(8 + U_0^0 - 4\ K_1 - K_1^\alpha\right)^2 + 8\left(4 + U_0^2 + U_1^0\right)\right] \quad . \quad (22)$$

Because differentiation of Eq. (22) with respect to either K_1 or K_1^α yields

$$4\,K_1 + K_1^\alpha = \frac{8 + U_0^0}{3}\,, \tag{23}$$

these two quantities cannot be completely determined. For convenience K_1 is given the value 1/3 used in the previous section and the optimization equations are solved for K_1^α. The optimum values of U_0^0 and U_1^0 can be obtained by use of Eq. (23) and differentiation of Eq. (22).

The resulting optimal system is shown in Table 2 along with the system described in the sensitivity section. It may be seen that about a 3 percent improvement is gained in overall performance and that sensitivity to parameter variations has been reduced at a slight cost to deterministic performance.

Table 2

	U_0^0	U_1^1	K_1	K_1^α	J^0	S	J
Deterministic	-1	-1	0.333	0.00	8.0	51.7	11.1
Stochastic	-1.17	-1.02	0.333	0.943	8.1	45.7	10.8
Sensitivity	3.04	--	--	0.18	--	--	--

Sensitivity of the Optimization

A natural question of interest is how sensitive the performance of the system is to values of the parameters of the controller. If the controller is optimal, first order sensitivities are zero and second order sensitivities are of interest. In Table 2 the sensitivities of performance to changes in U_0^0 and K_1^α found by taking second derivatives of Eq. (22) are listed. Note that the deterministic controller has a value of U_0^0 about 0.2 away from optimal and a value of K_1^α about 1 away from optimal. Squaring these two quantities and multiplying by the appropriate sensitivities yields a cost for the use of the deterministic controller of 0.12 because of improper nominal control U_0^0 and 0.18 because of not being adaptive.

There are two reasons why the optimal nominal control might vary from the optimal deterministic control: One reason is the dual control effect in which the control is modified to aid estimation. A second reason is the stochastic effect in which the control is modified to avoid the effects of uncertainty. Since no information is provided about $\Delta\alpha$ by the first measurement and it is known after the second, the dual control effect is not present in this example; hence the change of the nominal from the deterministic optimal is due to the stochastic effect. Had $\Delta\beta$ been selected rather than $\Delta\alpha$ the dual control effect would have appeared.

CONCLUSIONS

Several useful measures of the sensitivity of the performance of an optimal control system with respect to plant and controller parameter uncertainties, disturbances, and measurement noise have been described in terms of a simple control example. Optimal modification of the controller to reduce such sensitivity was also described.

REFERENCES

1. R. A. Rohrer and M. Sobral, "Sensitivity Considerations in Optimal System Design," IEEE Trans., Vol. AC-10, No. 1, pp. 43-48 (January 1965).

2. W. G. Tuel, I. Lee, and P. M. De Russo, "Synthesis of Optimal Control Systems with Sensitivity Constraints," presented at 1966 IFAC Conference, London, England, June 1966.

3. H. D'Angelo, M. L. Moe, and F. G. Hendricks, "Trajectory Sensitivity of an Optimal Control System," presented at Fourth Allerton Conference on Circuit and System Theory, University of Illinois, October 1966.

4. L. Meier, J. Peschon, B. Ho, R. Larson, and R. Dressler, "Design of Guidance and Control Systems for Optimum Utilization of Information," Final Report, Contract NAS2-3476, SRI Project 5967, Chapter III, pp. 25-44, Stanford Research Institute, Menlo Park, California (May 1967).

5. B. Pagurek, "Sensitivity of the Performance of Optimal Linear Control Systems to Parameter Variations," IEEE Trans., Vol. AC-10, pp. 178-180 (April 1965).

DOMINATING SET OF DIRECTED GRAPHS

YUMIN FU
Information and Control Science Center, Institute of Technology,
Southern Methodist University, Dallas, Texas 75222

ABSTRACT

A dominating set D of a directed graph G is defined as a subset of the vertex set of G such that every vertex not in D is the endpoint of some edge from a vertex in D. A minimal dominating set is a dominating set such that no subset has this property. In this paper, the existence of a minimal dominating set is discussed for both cyclic and acyclic graphs. A necessary and sufficient condition for a finite cyclic graph to have a dominating set D such that its complement $\overline{D}$ is also a dominating set is found.

This work was sponsored by the Air Force Office of Scientific Research, Office of Aerospace Research, United States Air Force, under Contract F44620-68-C-0023.

DISTINGUISHING CHARACTERISTICS OF OPTIMUM TREE

JOSEPH E. BARBAY AND GEORGE W. ZOBRIST
Department of Electrical Engineering
University of Missouri, Columbia, Missouri

ABSTRACT

The optimum tree for input table formulation for use with the program NASAP is defined as the tree which results in a flowgraph with the minimum number of loops to be found. Two distinguishing characteristics of this tree are developed for linear networks.

SUMMARY

Input tables for the network analysis program NASAP are formulated by using a tree of the network under consideration to categorize the elements of the network. Tree selection for this purpose influences the running time of the program and affects the maximum number of loops which must be stored. An optimum tree is defined as the tree which reduces, to a minimum, the number of loops to be found and stored. The general flowgraph's anatomy for linear networks is reviewed and the portions of the flowgraph which can influence the number of loops to be found and which can be affected by the tree selection is indicated.

The permanent of the connection matrix [C] for the closed flowgraph gives the number of terms present in the fully expanded determinant of the connection matrix [C]. Two features of the matrix [C] affect the value of the permanent of [C]. The number of nonzero terms included in [C] and the position of the nonzero terms in [C] determine the number of nonzero products in the fully expanded determinant of [C] which in effect determines the number of loops.

INTRODUCTION

NASAP is a computer program that applies a flowgraph approach to finding a transfer function between specified network variables. The input table is constructed by selecting a tree and corresponding cotree to use as an aid in classifying the network elements into different categories. Reference [1] deals more completely with the construction of the input table. The input table fully describes a flowgraph which represents the network and transfer function to be found. This flowgraph may vary in complexity for the same network depending upon the tree that was selected to form the input table. The difference in the degree of flowgraph complexity can range from flowgraphs which can be reduced to a transfer function to flowgraphs that cannot be reduced to a transfer function by NASAP, due to the limited number of loops which can be stored.

A dependent test source is connected to the network to indicate the transfer function to be found. A flowgraph of a network with such a dependent test source is called a "closed flowgraph". The specified

---

This work was supported by the National Aeronautics and Space Administration, Grant NGR 26-004-(035).

transfer function is obtained by finding the loops of the closed flowgraph by an exhaustive search routine and forming various loop products of disjointed loops. The loops and loop products are referred to as loops; the traceable loops of the flowgraph are distinguished as the "first order loops", while the loop products are known as "higher order loops". The number of loops to be evaluated depends upon the complexity of the flowgraph which in turn depends on the tree selected to form the input table. It is desirable to use a tree that results in the simplest flowgraph for the particular problem in question, thereby minimizing the running time of the program and the number of loops to be stored. An optimum tree can be defined as the tree or trees which, when used to categorize the elements of a network for the purpose of forming the input table, results in a flowgraph with the minimum number of loops to be found. This paper develops two distinguishing characteristics of this optimum tree for linear networks.

FLOWGRAPH DIFFERENCES

A general method of flowgraph formulation for linear, bilateral, passive networks can be found in reference [2]. This method can be extended to include networks with dependent sources by first treating the dependent sources as independent sources and forming the flowgraph by using the general method. Then edges, which represent the dependencies of the sources are added to the flowgraph. Figure 1-a shows a network with a dependent voltage controlled current source. Figures 1-b and 1-c are two linear trees of the network which are used to formulate the flowgraphs shown in figures 1-e and 1-f, respectively. The flowgraph in figure 1-e has 17 loops while the flowgraph of figure 1-f has 149 loops. When comparing the two flowgraphs to find possible differences which could account for the differences in the number of loops to be found, it is seen that the number of nodes and the allocation of the nodes between branch and link space is constant for both flowgraphs, since the number of branches and links of a given network is constant. The number of edges which correspond to dependent sources to be added is always the same since the number of dependent sources of the network does not change for different trees. The number of edges crossing from branch space to link space, representing Kirchhoff's voltage laws, and the number of edges crossing from link space to branch space, representing Kirchhoff's current laws, differs for the two flowgraphs. It is also possible to have flowgraphs for the same network with the same number of cross connecting edges connected in a different pattern in the flowgraphs. The position of the additional edges added to indicate the dependencies of the dependent sources can vary. Lastly, the labelling of the two flowgraph's nodes is different. The differences which can affect the number of loops to be found are:

(1) the number of cross connections between branch and link space and between link and branch space

(2) the position of the cross connecting edges and the position of the edges which represent the dependent sources

NUMBER OF LOOPS

A method of calculating the number of loops to be found when reducing a flowgraph to a transfer function is contained in reference [3]. The number of loops is equal to the permanent of the connection matrix [C], symbolized as $\overset{+}{|}C\overset{+}{|}$. The permanent of [C] is just the number of nonzero products in the expansion of det [C] when the expansion of det [C] is defined by,

$$\det [C] = \sum_{p} (\text{SIGN } p)\, A_{1p1}A_{2p2} \cdot \cdot \cdot \cdot \cdot A_{NpN}$$

the summation being taken over all permutation of p for the indices 1, 2, 3, . . ., N. The connection matrix [C] is defined as:

$C_{ij}=1$ for $i=j$
$C_{ij}=1$, if there is an edge from node j to node i
0, if otherwise
for $i \neq j$

Minimizing the permanent of [C] for the closed flowgraph by minimizing the number of nonzero products in the summation which is defined by det [C] will minimize the number of loops to be found. Two features of [C] affect the value of $\overset{+}{|}C\overset{+}{|}$; the first being the number of nonzero terms in [C]. For each zero of [C] there are (N-1)! zeros in the defining summation of det [C], where N is the number of nodes of the flowgraph. Minimizing the number of nonzero terms of [C] maximizes the total number of zeros appearing in the expansion of [C]. The second feature of [C] which affects the value of $\overset{+}{|}C\overset{+}{|}$ is the pattern of the zero and nonzero terms of [C]. The preferred pattern is one which widely distributes the zeros in the product terms of the summation defined by det [C]. This insures utilization of the zeros in [C] in an effective manner in reducing the value of $\overset{+}{|}C\overset{+}{|}$. From this definition of det [C], it can be seen that if all the zeros of [C] were concentrated in any row, for example, the first row, the first term, A_{1p1}, of each of the products $A_{1p1}A_{2p2} \cdot \cdot \cdot \cdot \cdot A_{NpN}$ could be zero. None of the remaining terms $A_{2p2}\ A_{3p3} \cdot \cdot \cdot \cdot \cdot A_{NpN}$ could ever be a zero, thus there could never be any products in the expansion of det [C] with more than one zero term included in it. This is the most effective distribution of zeros since products with more than one zero waste the effect of the redundant zeros. The same could be said for any column. The preferred pattern is where the zeros of [C] are concentrated into rows and columns of [C].

QUALITY NUMBERS

There are two distinguishing features of the connection matrix for the flowgraph formulated from the optimum tree. The first is that it has a minimum number of nonzero terms, thus maximizing the total number of zeros appearing in the expansion of det [C]. The second feature is that the zeros of [C] are located in the most effective possible pattern in [C], thus insuring wide distribution of the zeros in the product terms of the expansion of det [C]. Two quality numbers which are a measure of these distinguishing characteristics will now be defined.

The first quality number, QS, is just twice the number of nonzero terms of the connection matrix. Let the number of ones in each row and column of [C] be called the row and column count numbers. The quality number, QS, is then defined as the sum of the row and column count numbers. The second quality number, QP, is a measure of the effectiveness of the location of the zeros in [C]. A concentration of the zeros of [C] into as few rows and columns as possible is the desired pattern. The row and column count numbers will reflect this concentration of zeros as low count numbers for the rows and columns with high numbers of zeros. The quality number, QP, is defined to be the product of the row and column count numbers. The connection matrix of the flowgraph formed by use of the optimum tree will have a minimum QS and QP number. Tree number one is an example of such an optimum tree.

Figure 2-a and 2-b show the connection matrix for the flowgraph of figure 1-e and 1-f, respectively. The node numbers are divided into node numbers which pertain to voltage nodes and node numbers which pertain to current nodes. These two groups are further divided into nodes which refer to links and to nodes which refer to branches. Figure 3 shows a partitioning of the connection matrix of figure 2-a into eight regions. This ordering of the nodes and partitioning can be done for any flowgraph formulated by the general method referred to earlier. Regions R3 and R6 will always contain a null matrix while regions R1 and R8 will always contain a unit matrix, and hence the number and position of the nonzero elements in these regions is always constant. Regions R2 and R7 contain the branch portion of the fundamental circuit matrix $[B_{f12}]$ and $[B_{f12}]_t$ respectively, where $[B_f]=[B_{f11}\ B_{f12}]$ is defined by:

$b_{ij}=1$ if the jth element is included in the ith circuit
$b_{ij}=0$ if otherwise

The number of nonzero entries in region R2 and R7 is the number of ones in the branch portion of the fundamental circuit matrix, $[B_{f12}]$. These two regions are the only portions of the connection matrix in which the number of nonzero entries can vary for flowgraphs formulated from different trees. To minimize QS it is sufficient to minimize the number of nonzero entries of $[B_{f12}]$. The total number of ones included in $[B_{f12}]$ corresponds to the total number of branches contained in the circuits defined by the selected tree and corresponding cotree. A distinguishing characteristic of the optimum tree is the total number of branches included in the circuits defined by the optimum tree and corresponding cotree. The total number of branches included in the circuits defined by an optimum tree and cotree is a minimum.

The position of the nonzero entries of regions R2, R7, R4, and R5 can vary. There are two nonzero entries in each row of the regions R4 and R5. One of these nonzero entries in each row must be on the principal diagonal of [C], while the other nonzero entry can be in any column of the row. The nonzero entries of regions R2 and R7 can vary in position within the rows and columns of the regions R2 and R7. Each row and column of the regions R2 and R7 will always have at least one entry. To concentrate the zeros of [C] into rows, only the regions R2 and R7 need be considered since no row variation can take place outside of these regions. The number of ones in each row of region R2 is just the number of branches of the network graph included in the circuit corresponding to that row. To concentrate zeros in the rows of region R2 the circuits defined by the tree and corresponding cotree should have as few branches included in them as possible. The number of ones in each row of region R7 is just the number of circuits in the branch which corresponds to the row. To concentrate zeros in the rows of region R7 the branches of the selected tree should have as few circuits passing through them as possible. Concentration of zeros in rows of regions R2 and R7 concentrates the zeros in the columns of regions R7 and R2 respectively. Further concentration of the zeros in columns will result if the dependent voltage sources are dependent upon branch voltages of branches which have the most circuits included in them and the dependent current sources are dependent upon link currents of links which are included in the circuits which contain the greatest number of branches.

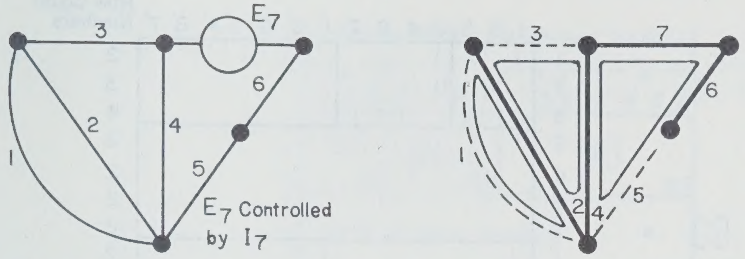
Figure 4 shows two additional trees of the network of figure 1-a. Tree number one has the lowest OS and OP numbers, therefore, the flowgraph formulated using tree number one has the least number of loops. This is shown by the table of figure 4. Tree number two has the highest OS and OP numbers of the four trees considered. The flowgraph formulated using tree number two has the greatest number of loops to be found and stored. Trees number three and four have the same OS number. Tree number four's OP number is smaller than that of tree number three, thus the number of flowgraph loops to be found and stored for tree number four is smaller than for tree number three.

CONCLUSION

The two distinguishing characteristics of the optimum tree are the two tree quality numbers, OS and OP. An optimum tree, for the purpose of formulating an input table for the program NASAP, will have a minimum OS and OP number. Choosing trees which have a minimum total number of branches included in the circuits defined by the tree and corresponding cotree will minimize the tree quality number OS. To minimize OP, each circuit of the fundamental circuit matrix should contain as few branches as possible and each branch of the tree should contain as few circuits as possible.

REFERENCES

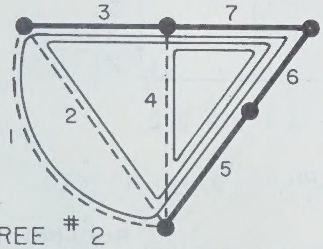
1. HAPP, W. W., "Flowgraphs as a Teaching Aid", IEE TRANSACTIONS ON EDUCATION, vol. E-9, no. 2, pp. 69-80, June, 1966.
2. MASON, S. J. and ZIMMERMANN, H. J., ELECTRONIC CIRCUITS, SIGNALS AND SYSTEMS, JOHN WILEY, New York, 1960.
3. JACOB, J. P., "The Number of Terms in the General Gain Formulas for Coates and Mason Signal Flowgraph", IEE TRANSACTION ON CIRCUIT THEORY, Vol. ct-12, no. 4, pp. 601-604, December, 1965.



E7
3
2
1
4
6
5
E7 Controlled
by I7
3
7
6
1
2
4
5

(a) NETWORK

(b) TREE # 1

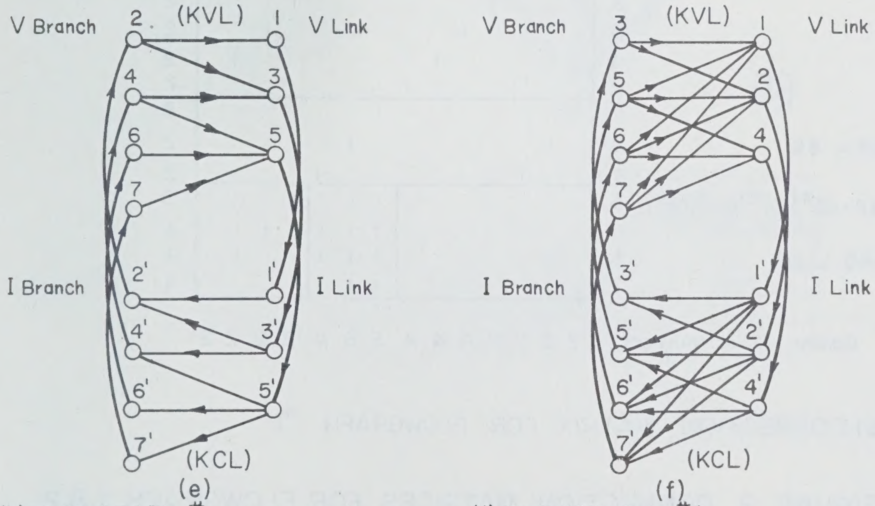


(c) TREE # 2

$$\left[B_f\right] = \begin{array}{c|ccc|cccc} & 1 & 3 & 5 & 2 & 4 & 6 & 7 \\ \hline 1 & 1 & & & 1 & & & \\ 3 & & 1 & & 1 & 1 & & \\ 5 & & & 1 & & 1 & 1 & 1 \end{array}$$

$$\left[B_f\right] = \begin{array}{c|ccc|cccc} & 1 & 2 & 4 & 3 & 5 & 6 & 7 \\ \hline 1 & 1 & & & 1 & 1 & 1 & 1 \\ 2 & & 1 & & 1 & 1 & 1 & 1 \\ 4 & & & 1 & & 1 & 1 & 1 \end{array}$$

(d) FUNDAMENTAL CIRCUIT MATRICES



V Branch
(KVL)
V Link
2
1
4
3
6
5
7
I Branch
I Link
2'
1'
4'
3'
6'
5'
7'
(KCL)
(e)
V Branch
(KVL)
V Link
3
1
5
2
6
4
7
I Branch
I Link
3'
1'
5'
2'
6'
4'
7'
(KCL)
(f)

(e) FLOWGRAPH # 1

(f) FLOWGRAPH #2

FIGURE 1. NETWORK, TWO TREES, TWO FUNDAMENTAL CIRCUIT MATRICES, and CORRESPONDING FLOWGRAPHS

QS = 66

QP = $(3^4)(2^{20})(3^2)(4^2)$

17 Loops

$$[C] = \begin{array}{c|ccccccc|ccccccc|c}
 & 1 & 3 & 5 & 2 & 4 & 6 & 7 & 1' & 3' & 5' & 2' & 4' & 6' & 7' & \text{Row Count Numbers} \\
\hline
1 & 1 & & & 1 & & & & & & & & & & & 2 \\
3 & & 1 & & 1 & 1 & & & & & & & & & & 3 \\
5 & & & 1 & & 1 & 1 & 1 & & & & & & & & 4 \\
\hline
2 & & & & 1 & & & & & & & 1 & & & & 2 \\
4 & & & & & 1 & & & & & & & 1 & & & 2 \\
6 & & & & & & 1 & & & & & & & 1 & & 2 \\
7 & & & & & & & 1 & & & & & & & 1 & 2 \\
\hline
1' & 1 & & & & & & & 1 & & & & & & & 2 \\
3' & & 1 & & & & & & & 1 & & & & & & 2 \\
5' & & & 1 & & & & & & & 1 & & & & & 2 \\
\hline
2' & & & & & & & & 1 & & & 1 & & & & 2 \\
4' & & & & & & & & & 1 & 1 & & 1 & & & 3 \\
6' & & & & & & & & & 1 & 1 & & & 1 & & 3 \\
7' & & & & & & & & & & 1 & & & & 1 & 2 \\
\hline
\text{Column Count Numbers} & 2 & 2 & 2 & 3 & 3 & 2 & 2 & 2 & 3 & 4 & 2 & 2 & 2 & 2 &
\end{array}$$

(a) CONNECTION MATRIX FOR FLOWGRAPH #1

QS = 86

QP = $(5^4)(2^{20})(3^2)(4^2)$

149 Loops

$$[C] = \begin{array}{c|ccccccc|ccccccc|c}
 & 1 & 2 & 4 & 3 & 5 & 6 & 7 & 1' & 2' & 4' & 3' & 5' & 6' & 7' & \text{Row Count Numbers} \\
\hline
1 & 1 & & & 1 & 1 & 1 & 1 & & & & & & & & 5 \\
2 & & 1 & & 1 & 1 & 1 & 1 & & & & & & & & 5 \\
4 & & & 1 & & 1 & 1 & 1 & & & & & & & & 4 \\
\hline
3 & & & & 1 & & & & & & & 1 & & & & 2 \\
5 & & & & & 1 & & & & & & & 1 & & & 2 \\
6 & & & & & & 1 & & & & & & & 1 & & 2 \\
7 & & & & & & & 1 & & & & & & & 1 & 2 \\
\hline
1' & 1 & & & & & & & 1 & & & & & & & 2 \\
2' & & 1 & & & & & & & 1 & & & & & & 2 \\
4' & & & 1 & & & & & & & 1 & & & & & 2 \\
\hline
3' & & & & & & & & 1 & 1 & & 1 & & & & 3 \\
5' & & & & & & & & 1 & 1 & 1 & & 1 & & & 4 \\
6' & & & & & & & & 1 & 1 & 1 & & & 1 & & 4 \\
7' & & & & & & & & 1 & 1 & 1 & & & & 1 & 4 \\
\hline
\text{Column Count Numbers} & 2 & 2 & 2 & 3 & 4 & 4 & 4 & 5 & 5 & 4 & 2 & 2 & 2 & 2 &
\end{array}$$

(b) CONNECTION MATRIX FOR FLOWGRAPH #1

FIGURE 2. CONNECTION MATRICES FOR FLOWGRAPH 1 & 2

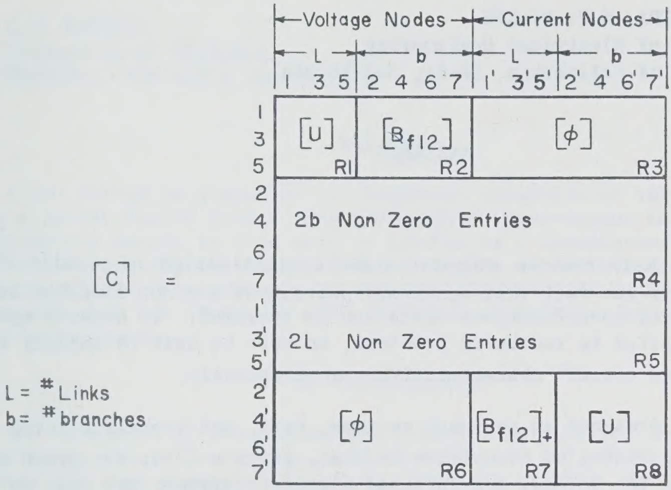
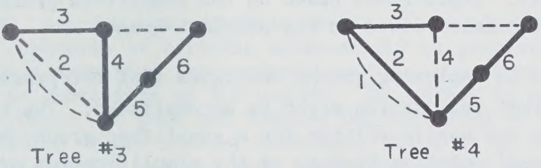

Voltage Nodes
Current Nodes
L
b
L
b
1 3 5
2 4 6 7
1' 3' 5'
2' 4' 6' 7'
1
3
5
2
4
6
7
1'
3'
5'
2'
4'
6'
7'
[U]
[B f12]
[φ]
R1
R2
R3
2b Non Zero Entries
R4
2L Non Zero Entries
R5
[φ]
[B f12]+
[U]
R6
R7
R8
[C] =
L = # Links
b = # branches

FIGURE 3. PARTITION CONNECTION MATRIX


3
2
1
4
6
5
Tree #3
3
2
1
4
6
5
Tree #4

Tree #	QS	QP	Number of Flowgraph Loops
1	66	$81 \times (2^{20})(3^2)(4^2)$	17
2	86	$625 \times (2^{20})(3^2)(4^2)$	149
3	70	$324 \times (2^{20})(3^2)(4^2)$	29
4	70	$225 \times (2^{20})(3^2)(4^2)$	21

FIGURE 4. TWO ADDITIONAL TREES WITH THE SAME QS NUMBER AND TREE COMPARISON TABLE.

GRAPH REALIZATION UTILIZING TREE, PATH, AND CIRCUIT LISTS

J. W. LAPATRA, K.D. WILLIS
Department of Electrical Engineering
University of California, Davis, California

ABSTRACT

One of the main reasons computer aided circuit design is usually iterative analysis is the fact that synthesis procedures are not flexible in the number and type of specifications which can be included. If network synthesis is to be applied to realistic problems, we must be able to satisfy a number of desired circuit characteristics simultaneously.

This paper presents an approach to tree, path, and circuit listing which allows algorithms to be written so that, given a list, the graph may be realized. In the method, conventional algebraic symbols are used to represent electric circuit connections. Expressions generated with this notation will contain additional special information which is useful in both enumeration and realization.

To accomplish enumeration, a method is given for the reduction of a complex graph to a simplified version for which the trees, paths, and circuits are easily listed. Expressions based on the simplified graph are then expanded to generate the total lists in the original graph.

Experience gained in analyzing graphs indicated that the procedure could be reversed so that realization might be accomplished. The tree, path, or circuit lists are easily written for a simplified graph; however, the lists contain unusual notation because of the simplification procedure. This fact may appear a disadvantage; however, the unusual terms may be removed by transformations. The advantage of the unusual forms is the fact that they are a unique characteristic of the simplified graph and with the list given in the unusual form, the simplified graph may be drawn immediately. The whole point of realization in this paper is that, given a tree, path, or circuit list, we will use the appropriate algorithm to prepare the list so that the simplified graph may be drawn, and then expanded.

The form of a list of graphical quantities from a simplified graph is sufficiently unique that an algorithm may be written capitalizing on the combinatorial qualities of the form. When one obtains the simplified graph representing the modified list, it is not difficult to return to the unsimplified graph.

ON THE SYNTHESIS OF RLC n-PORTS

G.O. MARTENS
University of Manitoba
Winnipeg, Manitoba, Canada

ABSTRACT

A new method is given for the algebraic insertion of variables to generate a parent hybrid matrix from a specified input-output matrix. The parent hybrid matrix is then used to synthesize a transformerless RLC n-port. It is shown that the state equations may easily be derived from the parent hybrid matrix and that the characteristic polynomial as well as the minimal polynomial of A are identical with the monic denominator polynomial of the input-output matrix.

I. INTRODUCTION

The synthesis of transformerless n-ports has received considerable attention in recent years. In particular, Boxall [1], Adams [2] and Olivares [3] have discussed RC n-port synthesis while D. Basson and C.C. Halkias [4] give a procedure for grounded RLC n-port synthesis which requires the partial fraction expansion of the immittance matrix. Cutteridge [2] borders an impedance matrix with polynomials containing unknown parameters and then inverts the bordered matrix to obtain a node admittance matrix. The resulting node admittance matrix, which contains complicated expressions in the unknown parameters, is realized as a grounded n-terminal network.

In this paper a new method is developed for algebraically inserting variables (converse of variable elimination) to generate a parent hybrid matrix from a specified input-output matrix. The parent hybrid matrix is then used to synthesize an RLC n-port. The method is not restricted to grounded n-ports and does not require the factoring of any polynomials or the calculation of any residue matrices. Also, it is shown that the state equations are easily derived from the parent hybrid matrix.

II. VARIABLE INSERTION METHOD

The input-output (transfer) matrix for an RLC n-port may be obtained from a parent matrix by the algebraic elimination of internal variables. At each stage of the elimination process, every 2×2 minor in the matrix of numerator polynomials is divisible by the common denominator polynomial [1,2,5]. This property will be referred to as the divisibility property.

The general method for variable insertion is described in Theorem 1. A symmetric matrix is here considered a degenerate case of a hybrid matrix. Polynomials in the differential operator p are denoted by lower case Latin letters, the coefficient of p^i in a polynomial b by b_i and the degree of a polynomial b by deg b.

Theorem 1:

Given a hybrid set of r linear simultaneous equations in matrix form:

$$\left[\begin{array}{c} v_1 \\ \vdots \\ v_t \\ \hline i_{t+1} \\ \vdots \\ i_r \end{array}\right] = \frac{1}{d_{11}} \left[\begin{array}{ccc|ccc} h_{11} & \cdots & h_{1t} & h_{1t+1} & \cdots & h_{1r} \\ \vdots & & \vdots & \vdots & & \vdots \\ h_{1t} & \cdots & h_{tt} & h_{tt+1} & \cdots & h_{tr} \\ \hline -h_{1t+1} & \cdots & -h_{tt+1} & h_{t+1t+1} & \cdots & h_{t+1r} \\ \vdots & & \vdots & \vdots & & \vdots \\ -h_{1r} & \cdots & -h_{tr} & h_{t+1r} & \cdots & h_{rr} \end{array}\right] \left[\begin{array}{c} i_1 \\ \vdots \\ i_t \\ \hline v_{t+1} \\ \vdots \\ v_r \end{array}\right] \tag{1}$$

or more compactly

$$Y_r = H_r X_r = \frac{1}{d_{11}} [h_{ij}] X_r \tag{2}$$

satisfying the conditions

(1) d_{11} and h_{ij} are polynomials in p with
deg $d_{11} = n$, deg $h_{ii} = n$ or $n + 1$, deg $h_{ij} \leq n$ $(i \neq j)$
and the leading coefficient of d_{11} is positive;

(2) d_{11} is relatively prime to h_{11} (or h_{rr});

(3) H_r satisfies the divisibility property
i.e., $d_{11} \mid (h_{kk}h_{ij} - h_{ik}h_{kj})$ for $i,j,k = 1,2,\ldots,r$.

Then either a "current" or a "voltage" variable may be inserted such that if the inserted variable is eliminated the original matrix equation is obtained.

When a current variable i_o is inserted, the hybrid matrix equation

$$Y_{r+1} = H_{r+1} X_{r+1} = \frac{1}{d_{22}} [f_{ij}] X_{r+1} \tag{3}$$

with $Y_{r+1} = \begin{bmatrix} v_o \\ Y_r \end{bmatrix}$ and $X_{r+1} = \begin{bmatrix} i_o \\ X_r \end{bmatrix}$

is obtained satisfying the conditions

(1a) d_{22} and f_{ij} are polynomials in p with deg $d_{22} = n-1$, deg f_{ii} = deg $h_{ii} - 1$ $(i \neq 1)$, deg $f_{ij} \leq n-1$ $(i \neq j)$ and the leading coefficient of d_{22} is positive;

(2a) d_{22} is relatively prime to $f_{11} = d_{11}$;

(3a) H_{r+1} satisfies the divisibility property.

When a voltage variable v_{r+1} is inserted, the hybrid matrix equation

$$Y_{r+1} = H_{r+1} X_{r+1} = \frac{1}{d_{22}} [g_{ij}] X_{r+1} \tag{4}$$

with $Y_{r+1} = \begin{bmatrix} Y_r \\ i_{r+1} \end{bmatrix}$ and $X_{r+1} = \begin{bmatrix} X_r \\ v_{r+1} \end{bmatrix}$

is obtained satisfying conditions (1a), (3a); and (2b) d_{22} is relatively prime to $g_{r+1r+1} = d_{11}$.

Proof:

It may be assumed that d_{11} and h_{11} are relatively prime. For if not, then d_{11} and h_{rr} are relatively prime and by permuting rows and corresponding columns, and with the appropriate change in notation, a new set of hybrid equations is obtained which satisfy the assumption.

Two lemmas are required to complete the proof of the theorem. The results of the first lemma and its corollary are similar to those given by Talbot [2] for cascade and Brune synthesis of driving point functions.

Lemma 1:

Given two relatively prime polynomials u and b with deg u = n, deg b = n+1 (or n), $u_n>o$ and $b_{n+1}>o$ (or $b_n>o$), then there exist polynomials d, c and e such that

$$db - cu = \mp e^2 \tag{5}$$

where deg d = n-1, deg c = deg b-1, deg $e \leq n-1$ and d is relatively prime to u. The sign in (5) can be chosen to make $d_{n-1}>o$ and $c_n>o$ (or $c_{n-1}>o$). Furthermore, once the polynomial e has been chosen, the polynomials d and c are uniquely determined.

Proof:

There exist unique polynomials h and g with deg h<n (or deg h<n-1) and deg g<n such that [6]

$$gb + hu = 1 \; . \tag{7}$$

Multiplying by $\mp e^2$ and using the division algorithm yields polynomials c and d, that is,

$$\mp e^2 g = qu + d$$

and

$$\pm e^2 g = \ell b + c$$

where the upper or lower sign is chosen to make $d_{n-1}>o$. The proof that e can be chosen to satisfy the remaining requirements of the lemma, and that d and c are unique, is given elsewhere [5].

Corollary 1:

Given u, b and e as defined in Lemma 1, and a polynomial v with deg $v \leq n$, then there exist unique polynomials m and z with deg $m \leq n$ and deg $z \leq n-1$ such that

$$zb + mu = ev \; . \tag{8}$$

Proof:

Starting with (7):

$$gb + hu = 1$$

and by the division algorithm:

$$evg = xu + z$$

and hence

$$m = xb + evh \; .$$

Lemma 1 is applied to d_{11} and h_{11} with $u = d_{11}$ and $b = h_{11}$ giving rise to two cases, A and B.

Case A:

The sign in (5) must be chosen negative to make the leading coefficient of d_{22} positive. Let $e = f_{12}$ and $d = d_{22}$. Corollary 1 is now applied with $v = h_{1j}$ determining $z = f_{1j+1}$ for $j = 2,3,\ldots,r$. An augmented set of equations may now be defined as follows:

$$\begin{bmatrix} i_o \\ \hline Y_r \end{bmatrix} = \frac{1}{d_{11}} \left[\begin{array}{c|c} d_{22} & -T_{12} \\ \hline T_{21} & H_r \end{array}\right] \begin{bmatrix} v_o \\ \hline X_r \end{bmatrix} \tag{9}$$

where $T_{12} = [f_{1j}]$, $T_{21} = [f_{j1}]$ $j = 2,3,\ldots, r+1$,
$f_{j1} = f_{1j}$ for $j = 2,3,\ldots, t+1$ and $f_{j1} = -f_{1j}$ for $j = t+2, t+3,\ldots,r+1$.
Some divisibility conditions are now required and these are stated in Lemma 2.

Lemma 2:

The following divisibility conditions corresponding to 2 x 2 minors of (9) are satisfied.

$$d_{11} \mid (f_{12}\, h_{ij} - f_{1j+1}\, h_{i1}) \tag{10}$$

$$d_{11} \mid (f_{21}\, h_{ij} - f_{i+1\,1}\, h_{1j}) \tag{11}$$

$$d_{11} \mid (f_{1\,j+1}\, h_{1k} - f_{1\,k+1}\, h_{1j}) \tag{12}$$

$$d_{11} \mid (f_{i+1\,1}\, h_{k1} - f_{k+1\,1}\, h_{i1}) \tag{13}$$

$$d_{11} \mid (d_{22}\, h_{ij} + f_{i+1\,1}\, f_{1\,j+1}) \tag{14}$$

where $i,j,k = 1,2,\ldots,r$.

Proof:

The divisibility conditions are proved in sequence, starting with (10). The details are given elsewhere [5].

If f_{11} is set equal to d_{11}, by (14) $f_{i+1\,j+1}$ can be defined by

$$f_{11} f_{i+1\,j+1} = d_{22}\, h_{ij} + f_{i+1\,1}\, f_{1\,j+1} \quad \text{for } i,j = 1,2,\ldots,r. \tag{15}$$

Condition (1a) can now, easily, be verified from (15).

If the first equation in (9) is solved for v_o and the result substituted into the second matrix equation of (9), then

$$\begin{bmatrix} v_o \\ \hline Y_r \end{bmatrix} = \frac{1}{d_{22}} \left[\begin{array}{c|c} d_{11} & T_{12} \\ \hline T_{21} & \frac{1}{d_{11}} (d_{22} H_r + T_{21} T_{12}) \end{array}\right] \begin{bmatrix} i_o \\ \hline X_r \end{bmatrix} \tag{16}$$

Using (15), equation (16) has the form of (3), that is, a current variable, i_o has been inserted. The steps leading from (9) to (16) and from (16) to (3) are all reversible, therefore the inserted variable may be eliminated yielding the original equation (2). Using (15) it may be shown [5] that

$$d_{22} \mid (f_{kk} f_{ij} - f_{ik} f_{kj})$$

thus verifying the divisibility property for H_{r+1}.

Case B:

The proof is similar to Case A except that the sign in (5) must be chosen positive and a variable is inserted.

Theorem 1 may now be applied, iteratively, to an input-output matrix which satisfies the divisibility property and has at least one numerator polynomial relatively prime to the denominator polynomial. The resulting matrix equation is of the form

$$\begin{bmatrix} V_1 \\ V_2 \\ \hline I_3 \\ I_4 \end{bmatrix} = \left[\begin{array}{cc|cc} pL_{11}+H_{11} & H_{12} & H_{13} & H_{14} \\ H_{12}^T & H_{22} & H_{23} & H_{24} \\ \hline -H_{13}^T & -H_{23}^T & H_{33} & H_{34} \\ -H_{14}^T & -H_{24}^T & H_{34}^T & pC_{44}+H_{44} \end{array}\right] \begin{bmatrix} I_1 \\ I_2 \\ \hline V_3 \\ V_4 \end{bmatrix} \tag{17}$$

where I_2 and V_3 are the specified independent port variables while I_1 and V_4 are the inserted independent port variables. The entries of all the submatrices are real and specifically, L_{11} and C_{44} are diagonal and positive definite. The parent hybrid matrix of Eq.(17) describes a resistive multiport with an inductance in series with each inserted current variable and a capacitor in parallel with each inserted voltage variable.

The RLC network is synthesized by first synthesizing the resistive multiport and then connecting the inductances and capacitances. The specified input-output matrix is then realized by short circuiting ($V_1 = 0$) the ports with the series inductances and open circuiting ($I_4 = 0$) the ports with parallel capacitances.

In the application of Lemma 1, the choice of polynomial $e = f_{12}$ (or $f_{r\ r+1}$) is required. Possible choices may be generated by using the divisibility property which requires that

$$f_{12}h_{12} - f_{13}h_{11} = m\ d_{11} \tag{18}$$

In fact, the division algorithm may be used to show that it suffices to find f_{12} and f_{13} satisfying

$$f_{12}k_{12}^{(1)} - f_{13}k_{11}^{(1)} = m^{(1)}d_{11} \tag{19}$$

where $k_{12}^{(1)}$ and $k_{11}^{(1)}$ are the remainders upon the division of $h_{12}^{(1)}$ and $h_{11}^{(1)}$ by d_{11}. The superscripts indicate that the common factors, if any, have been removed. In particular, $m^{(1)}$ may be chosen to be the zero polynomial. Also, in the application of Lemma 1, if d_{22} and f_{12} satisfy (5) then it is easily verified that $\lambda^2 d_{22}$ and λf_{12} satisfy (5). The real constant λ is chosen so that the absolute values of the transfer functions in the parent matrix do not exceed 1, as required for transformerless realizability.

III. DERIVATION OF THE STATE EQUATIONS FROM THE PARENT HYBRID MATRIX

The state equations are derived from the parent hybrid matrix as follows: Let V_1 and I_4 equal zero and rearrange Eq. (17) to yield

$$p\begin{bmatrix} I_1 \\ V_4 \end{bmatrix} = -\begin{bmatrix} L_{11}^{-1} & 0 \\ 0 & C_{44}^{-1} \end{bmatrix}\begin{bmatrix} H_{11} & H_{14} \\ -H_{14}^T & H_{44} \end{bmatrix}\begin{bmatrix} I_1 \\ V_4 \end{bmatrix} - \begin{bmatrix} L_{11}^{-1} & 0 \\ 0 & C_{44}^{-1} \end{bmatrix}\begin{bmatrix} H_{12} & H_{13} \\ -H_{24}^T & H_{34}^T \end{bmatrix}\begin{bmatrix} I_2 \\ V_3 \end{bmatrix}$$

$$\begin{bmatrix} V_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} H_{12}^T & H_{24} \\ -H_{13}^T & H_{34} \end{bmatrix}\begin{bmatrix} I_1 \\ V_4 \end{bmatrix} + \begin{bmatrix} H_{22} & H_{23} \\ -H_{23}^T & H_{33} \end{bmatrix}\begin{bmatrix} I_2 \\ V_3 \end{bmatrix} \tag{20}$$

Note that L_{11} and C_{44} are nonsingular since by construction they are diagonal positive definite. The above matrix equations are in the standard state equation form [7]:

$$px = Ax + Bu$$

$$y = Cx + Du$$

with the obvious identifications. As is well-known, the input-output matrix

$$H_r = C(pI - A)^{-1} B+D$$

and since the order of A is equal to the lowest common denominator polynomial of H_r, there cannot be any cancellations in $(pI - A)^{-1}$. Therefore, the characteristic polynomial of A, the minimal polynomial of A and the monic common denominator polynomial of H_r are all identical.

EXAMPLE 1:

Given:

$$\left[\begin{array}{c} v_1 \\ \hline i_2 \\ i_3 \end{array}\right] = \frac{1}{d_{11}} \left[\begin{array}{c|cc} 6p^2+1 & -(6p^2+1) & 6p^3+6p^2+p+1 \\ \hline 6p^2+1 & 6p^3+4 & -3p \\ -(6p^3+6p^2+p+1) & -3p & 3p(p+1) \end{array}\right] \left[\begin{array}{c} i_1 \\ \hline v_2 \\ v_3 \end{array}\right]$$

$$d_{11} = 6p^3 + 6p^2 + 4p + 1$$

In Eq. (7), let $u = d_{11}$ and $b = h_{33}$, then [6]

$$g = \frac{-2}{9} (12p^2 + 9p + 8)$$

and

$$h = \frac{4}{3} p + 1 \quad .$$

Applying Eq. (19) yields

$$(-1)f_{34} - (p+1)f_{24} = 0 \quad .$$

Hence possible choices for f_{34} and f_{24} are $-(p+1)$ and 1, respectively. By Lemma 1

$$\mp \frac{f_{34}^2 g}{d_{11}} = q + \frac{d_{22}}{d_{11}}$$

hence, (choosing the upper sign)

$$\frac{(p+1)^2 \; 2/q \; (12p^2+qp+8)}{6p^3+6p^2+4p+1} = \frac{4}{9} p + \frac{7}{9} + \frac{2p^2+2p+1}{6p^3+6p^2+4p+1}$$

and so $d_{22} = 2p^2 + 2p + 1$.

Similarly, using Lemma 1

$$f_{33} = p + 1 \quad .$$

Applying Corollary 1 yields $f_{14} = 1$ and $f_{ij} = \dfrac{d_{22}\, h_{ij} + f_{i4} f_{4j}}{d_{11}}$ for i,j = 1, 2,3 (similar to (15)).

A voltage variable, v_4 has now been inserted:

$$\left[\begin{array}{c} v_1 \\ \hline i_2 \\ i_3 \\ i_4 \end{array}\right] = \frac{1}{d_{22}} \left[\begin{array}{c|ccc} 2p & -2p & 2p^2+2p & 1 \\ \hline 2p & 2p^2+4p+1 & -1 & 1 \\ -(2p^2+2p) & -1 & p+1 & -(p+1) \\ -1 & 1 & -(p+1) & 6p^3+6p^2+4p+1 \end{array}\right] \left[\begin{array}{c} i_1 \\ \hline v_2 \\ v_3 \\ v_4 \end{array}\right]$$

$$d_{22} = 2p^2 + 2p + 1 \quad .$$

Similarly, v_5 and i_6 are inserted to yield

$$\begin{bmatrix} v_6 \\ v_1 \\ \hline i_2 \\ i_3 \\ i_4 \\ i_5 \end{bmatrix} = \left[\begin{array}{cc|cccc} 2p & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 & 0 & 1 \\ -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 3p & 0 \\ -1 & -1 & 1 & 0 & 0 & p+1 \end{array}\right] \begin{bmatrix} i_6 \\ i_1 \\ \hline v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix}$$

The state equation coefficient matrices are obtained by letting $v_6 = i_4 = i_5 = 0$ and rearranging the above equations as in Eq. (20):

$$A = -\Lambda^{-1} \left[\begin{array}{c|cc} 0 & -1 & 1 \\ \hline 1 & 0 & 0 \\ -1 & 0 & 1 \end{array}\right] \quad B = -\Lambda^{-1} \left[\begin{array}{c|cc} 0 & 0 & 1 \\ \hline 0 & 0 & 0 \\ -1 & 1 & 0 \end{array}\right] \quad \text{where } \Lambda = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$C = \left[\begin{array}{c|cc} 0 & 0 & 1 \\ \hline 0 & 0 & 1 \\ -1 & 0 & 0 \end{array}\right] \quad D = \left[\begin{array}{c|cc} 0 & 0 & 1 \\ \hline 0 & 1 & 0 \\ -1 & 0 & 0 \end{array}\right]$$

An additional current variable may be inserted by inspection:

$$\begin{bmatrix} v_7 \\ v_6 \\ v_1 \\ \hline i_2 \\ i_3 \\ i_4 \\ i_5 \end{bmatrix} = \left[\begin{array}{ccc|cccc} 1 & 0 & 0 & -1 & 0 & 0 & -1 \\ 0 & 2p & 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 3p & 0 \\ 1 & -1 & -1 & 0 & 0 & 0 & p \end{array}\right] \begin{bmatrix} i_7 \\ i_6 \\ i_1 \\ \hline v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix}$$

The above hybrid matrix is realized by the network of Figure 1.

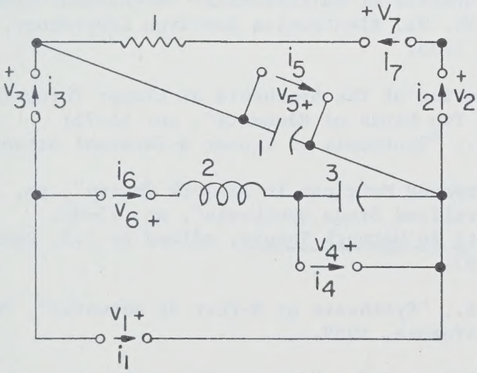


FIGURE 1: NETWORK REALIZING THE HYBRID MATRIX OF EXAMPLE 1.

The input-output matrix is realized when $i_4 = i_5 = v_6 = v_7 = 0$.

IV. CONCLUSIONS

A new method has been introduced for inserting variables algebraically to generate a parent hybrid matrix from a specified input-output matrix which satisfies the divisibility property and has at least one numerator polynomial relatively prime to the common denominator polynomial. In particular, the method has the advantage that it does not require the factoring of any polynomials nor the calculation of any residue matrices and is as readily applicable to a multiport as to a one-port. Although RLC realizability is not guaranteed for every choice of the polynomial e in Eq.(5), the proper choice of e at each stage leads to a realization provided the input-output matrix is realizable with n reactive elements (where n is the degree of the denominator polynomial) plus any number of resistors.

Another advantage of the method is that it leads to state equations which are in the proper form required for RLC networks as compared to the usual methods which lead to canonical matrices. Furthermore, the characteristic polynomial of A, the minimal polynomial of A and the monic common denominator polynomial of the input-output matrix are identical, ensuring that no spurious modes have been introduced.

ACKNOWLEDGEMENT

The author wishes to acknowledge the advice and encouragement of the late Professor S. Seshu. He also wishes to express his thanks to Professors M.E. Van Valkenburg, J.B. Cruz, Jr., and G.A. Metze for their comments and valuable suggestions during this investigation. This work was supported in part by Joint Services Electronics Program Contract DA28043 AMC 00073(E) and in part by National Science Foundation Grant NSF GK36 while the author was with the Coordinated Science Lab., University of Illinois, Urbana, Ill.

REFERENCES:

[1] Boxall, F.S., "Synthesis of Multiterminal Two-Element-Kind Networks", Technical Report No. 95, Electronics Research Laboratory, Stanford University, Nov., 1955.

[2] Adams, K.M., "A Review of the Synthesis of Linear Three-Terminal Networks Composed of Two Kinds of Elements", pp. 65-75;
Cutteridge, O.P.D., "Synthesis of Linear N-Terminal Networks", pp. 121-131;
Bryant, P.R., "Compound Matrices in Network Theory", pp. 3-18;
Talbot, A., "Generalized Brune Synthesis", pp. 75-90;
Recent Developments in Network Theory, edited by S.R. Deards, The Macmillan Co., 1963.

[3] Olivares, Jr., J.E., "Synthesis of N-Port RC Networks", Ph.D. Thesis, University of California, 1959.

[4] Basson, D. and Halkias, C.C., "Canonic RLC N-Port Networks", Journal of the Franklin Institute, Vol. 282, No. 2, pp. 65-84, August, 1966.

[5] Martens, G.O., "Algebraic Generation and Active Network Realization of State Equations", Ph.D. Thesis, Report R-294, Coordinated Science Lab., University of Illinois, Urbana, Ill., May, 1966.

[6] Bocher, M., Introduction to Higher Algebra, Dover Publications, Inc., pp. 188-202, 1964.

[7] Zadeh, L.A., and Desoer, C.A., Linear System Theory, McGraw-Hill Book Co., Inc., 1963.

THE DETERMINATION OF PORT-TREE STRUCTURES OF TRANSFORMERLESS (n+1)-TERMINAL n-PORT NETWORKS FROM THEIR SHORT-CIRCUIT ADMITTANCE PARAMETER MATRICES

DEAN J. SCHMIDLIN AND FRANK J. LUPO
New York University, Department of Electrical Engineering,
Bronx, New York, 10453

ABSTRACT

This paper presents a technique for determining all possible port tree-structures of (n+1)-terminal n-port networks from their short-circuit admittance parameter matrices $Y(s)$. First the complex frequency variable s in $Y(s)$ is set equal to some real positive value σ. From $Y(\sigma)$ port orientation matrices P are determined. Then all port tree-structures on which $Y(s)$ are realizable are synthesized from the P matrices. A computer program implementation ("POM") of this technique has been written.

I. INTRODUCTION

Many short-circuit admittance parameter matrices are realizable as transformerless (n+1)-terminal n-port networks. The set of port orientation matrices P that correspond to such an admittance matrix $Y(s)$ specifies all of the possible one-tree port-structures which realize $Y(s)$. In this paper a technique is presented for determining all of the port orientation matrices from the given short-circuit admittance parameter matrix. From these P-matrices one may determine by well known methods all of the corresponding port-trees.[1-6]

II. DEFINITIONS AND BASIC THEOREMS

The port orientation matrix P of a transformerless (n+1)-terminal n-port network specifies completely the relative orientations of the ports in its associated port-tree. Each element p_{ij} in P is defined as follows: $p_{ij} = +1$ if the tree path directions of ports i and j in the port-tree are confluent. $p_{ij} = -1$ if the tree path directions of ports i and j in the port-tree are counterfluent.

Two ports in a port-tree are in series if they are incident at a common node of degree two. A tree of series ports will be called a linear tree. A node of degree one in a port-tree will be called a tip node, and the port incident at that node will be called a tip port. If a pair of tip ports are incident at a common node, these tip-ports will be referred to as star-connected tip ports.

The following three theorems are concerned with the elements in the P-matrix corresponding to a conductance matrix G.

THEOREM 1.[5] If a real symmetric matrix G is the short-circuit conductance matrix of a transformerless (n+1)-terminal n-port network having all positive branch values and element g_{ij} in G is non-zero, then

$$p_{ij} = +1 \quad \text{if} \quad g_{ij} > 0$$

$$p_{ij} = -1 \quad \text{if} \quad g_{ij} < 0$$

THEOREM 2.[6] If ports i and j are in series then one must have

$$p_{ij} = p_{ki}\, p_{kj} \quad \text{for all} \quad k \neq i, j \tag{1}$$

On the other hand, if (1) is satisfied for two ports i and j, then they are either in series or contained in a linear port subtree.

THEOREM 3.[6] If ports i and j are star-connected tip ports then

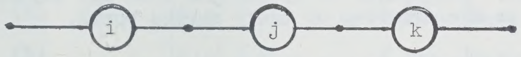
$$p_{ij} = -\, p_{ki}\, p_{kj} \quad \text{for all} \quad k \neq i, j \tag{2}$$

and conversely, if (2) is satisfied for two ports i and j, then ports i and j are star-connected tip ports.

The next two theorems are helpful in the determination of tip ports.

THEOREM 4. If g_{ij} is the only zero element in row i of matrix G then port j is a tip port.

Proof: Assume that port j is not a tip port. Then there must exist a port k such that ports i, j, and k form the following linear subtree when all of the other ports are short-circuited



In order that g_{ij} be zero in this port-tree, g_{ik} must be zero also. But this contradicts the hypothesis that g_{ij} is the only zero element in row i of G. Consequently port j must be a tip port.

THEOREM 5.[6] If matrix G is realizable as a transformerless (n+1)-terminal n-port network composed of all nonnegative branches and if there is an off-diagonal element in G whose magnitude is distinctly less than any other element, then the ports that correspond to this element are tip ports.

III. RELATIVE ORIENTATION OF TWO PORTS HAVING ZERO MUTUAL ADMITTANCE

The port orientation matrix P that corresponds to an admittance matrix Y(s) may be determined from its corresponding conductance matrix G, where $G = Y(s) \mid s = \sigma$ and where σ is real and positive. If admittance matrix Y(s) is realizable as a transformerless (n+1)-terminal n-port RLC network it is realizable on the same port-tree structure as G, and consequently both have the same P matrices.

If the ports of the network are redefined so that its new port-structure is a star-tree then its new G-matrix will be hyperdominant and its new Y(s)-matrix will contain the negative of p.r. functions in its off-diagonal positions and the following p.r. function in its main diagonal position,

$$y_{ii}(s) = y'_{ii}(s) - \sum_{\substack{j=1 \\ j \neq i}}^{n} y_{ij}(s)$$

where $y'_{ii}(s)$ is the value of the admittance that is connected directly across port i. Consequently, the new Y(s) will be realizable as a passive RLC n-port network on that port tree-structure.

The elements in matrix P that correspond to the non-zero elements in the matrix G may be determined by Theorem 1. To complete the specification of matrix P one must determine the value of its elements that correspond to the zero elements in G. This may be done as follows. Consider a 4-th order submatrix G_1 of G in which g_{km} is a zero element and submatrix P_1 of P in which p_{km} is an unknown. Since these matrices are symmetric their lower triangular terms may be omitted for convenience, i.e.

$$G_1 = \begin{array}{c} \\ k \\ i \\ j \\ m \end{array} \begin{array}{cccc} k & i & j & m \\ \hline g_{kk} & g_{ki} & g_{kj} & 0 \\ & g_{ii} & g_{ij} & g_{im} \\ & & g_{jj} & g_{jm} \\ & & & g_{mm} \end{array} \qquad P_1 = \begin{array}{c} \\ k \\ i \\ j \\ m \end{array} \begin{array}{cccc} k & i & j & m \\ \hline 1 & p_{ki} & p_{kj} & p_{km} \\ & 1 & p_{ij} & p_{im} \\ & & 1 & p_{jm} \\ & & & 1 \end{array}$$

where G_1 is the conductance matrix that results when all ports but k, i, j and m are short-circuited. If P_1 cannot be uniquely determined from G_1 then not all of the port-subtrees upon which G_1 is realizable may be valid port subtrees of G. Thus the correct value of p_{km} can be obtained assuredly only if P_1 is unique. Uniqueness of P_1 assures that all port-subtrees realizing G_1 differ only in the ordering of their series ports.

The following theorem is for an important special case in which the upper portion of P_1 may contain two unknown elements.

THEOREM 6. If G_1 is a short-circuit conductance matrix that is realizable as a 5-terminal 4-port network consisting of ports k, i, j, and m, and if g_{ij} and g_{km} are the only zero elements in the upper portion of G_1 then the port-tree realizing G_1 is a star and elements p_{ij} and p_{km} in P_1 are given by,

$$p_{ij} = - p_{ki}\, p_{kj} = - p_{im}\, p_{jm}$$

$$p_{km} = - p_{ki}\, p_{im} = - p_{kj}\, p_{jm}$$

Proof: Since g_{ij} and g_{km} are the only zero elements in the upper portion of G_1, there exists only one zero in each row of G_1. From Theorem 4, ports k, i, j, and m are tip ports and consequently the port-tree is a star. The values of p_{ij} and p_{km} are then given by Theorem 3.

Except for the special case given in Theorem 6, all elements in P_1 other than p_{km} must be known. The following theorem is the result of a complete investigation of all 4-th order submatrices G_1 that have a unique port orientation matrix P_1 (a detailed proof of this theorem is given in Reference 6).

THEOREM 7. If G_1 is a short-circuit conductance matrix that is realizable as a 5-terminal 4-port network consisting of ports k, i, j, and m, and if element p_{km} is the only unknown element in its unique port orientation matrix P_1, then p_{km} may be determined from,

(1) $$p_{ki}\ p_{kj} = p_{im}\ p_{jm} = p_{ij}$$

	CONDITIONS	p_{km}
(a)	$\lvert g_{ij}\rvert < M_1$ OR $< M_2$	$+\ p_{ki}\ p_{im}$
(b)	$\lvert g_{ij}\rvert > M_1$ AND $> M_2$	
	$m_1 = m_2$	$-\ p_{ki}\ p_{im}$
	$m_1 \neq m_2$	$+\ p_{ki}\ p_{im}$
(c)	$\lvert g_{ij}\rvert = M_1$ OR $= M_2$	
	$m_1 \neq m_2$	$+\ p_{ki}\ p_{im}$

(2) $$p_{ki}\ p_{kj} = p_{im}\ p_{jm} = -\ p_{ij}$$

CONDITIONS	p_{km}
g_{ki} OR $g_{kj} \neq 0$	
g_{im} OR $g_{jm} \neq 0$	$-\ p_{ki}\ p_{im}$

(3) $$p_{ki}\ p_{kj} = -\ p_{im}\ p_{jm} = p_{ij}$$

	CONDITIONS	p_{km}
(a)	$\lvert g_{ij}\rvert > M_1$	$-\ p_{km_1}\ p_{m_1 m}$
(b)	$\lvert g_{ij}\rvert < M_1$	$+\ p_{km_2}\ p_{m_2 m}$
(c)	$\lvert g_{ij}\rvert = M_1$	
	$g_{m_1 m} \neq 0$	$-\ p_{km_1}\ p_{m_1 m}$

(4) $$p_{ki}\ p_{kj} = -\ p_{im}\ p_{jm} = -\ p_{ij}$$

	CONDITIONS	p_{km}
(a)	$\lvert g_{ij}\rvert > M_2$	$-\ p_{km_2}\ p_{m_2 m}$
(b)	$\lvert g_{ij}\rvert < M_2$	$+\ p_{km_1}\ p_{m_1 m}$
(c)	$\lvert g_{ij}\rvert = M_2$	
	$g_{km_2} \neq 0$	$-\ p_{km_2}\ p_{m_2 m}$

WHERE

$M_1 = \text{MIN}\ (\lvert g_{ki}\rvert\ ,\ \lvert g_{kj}\rvert)$

$M_2 = \text{MIN}\ (\lvert g_{im}\rvert\ ,\ \lvert g_{jm}\rvert)$

$m_1 = i$ OR j IF $M_1 = \lvert g_{ki}\rvert$ OR $\lvert g_{kj}\rvert$ RESPECTIVELY

$m_2 = i$ OR j IF $M_2 = \lvert g_{im}\rvert$ OR $\lvert g_{jm}\rvert$ RESPECTIVELY

Depending on the relative values of the elements in submatrix P_1 the conditions of the theorem fall into four categories. If any of the conditions in the first column are satisfied then submatrix P_1 is unique and the value of p_{km} is given in the second column. In general, submatrix G_1 will contain zero elements other than g_{km} but the elements in P corresponding to these zeros shall have been determined previously.

Given that element p_{km} in the matrix P is unknown, the search for a unique P_1-submatrix may be performed as follows,

(1) Select an i and a j such that elements p_{ki}, p_{kj}, p_{im}, and p_{jm} in P are known.

(2) Apply Theorems 6 or 7 to the submatrices P_1 and G_1 formed by ports k, i, j, and m. If P_1 is unique then p_{km} is determined.

(3) Repeat the process until either a unique submatrix P_1 is found or until there are no more i's and j's that satisfy (1).

IV. PROCEDURE FOR DETERMINING P-MATRICES

The procedure for determining all of the port orientation matrices from a short-circuit admittance parameter matrix Y(s) is divided into five parts:

(1) Formation of conductance matrix $G = Y(s)|_{s = \sigma}$ where σ is real and positive.

(2) Determination of the elements in P that correspond to the non-zero elements in G.

(3) Determination of the elements in P that are constrained by the non-zero elements in G.

(4) Assignment of a set of q independent elements and determination of the remaining unknown elements in P.

(5) Determination of the remaining 2^q-1 port orientation matrices.

The second part of the procedure is accomplished by applying Theorem 1. Assuming that parts (1) and (2) have been completed, part (3) is performed as follows:

Step 1 Select an unknown element in the upper triangular portion of the matrix P. Search for a unique submatrix P_1 containing this element and thereby determine its value. If a unique submatrix P_1 cannot be found neglect this element until additional elements in the same row or column are determined.

Step 2 Repeat step 1 until either all unknown elements are determined or until the application of step 1 fails.

If after the completion of part (3), there are no unknown elements in the matrix P then P is unique and the procedure is terminated. If unknown elements remain in matrix P then P is not unique and a set of independent elements must be found.

A set of q independent elements is determined in part (4) which consists of the following steps:

Step 1 Select an unknown element in P such that the assignment of +1 to this element constrains at least one other element in the same row or column. If no such element exists then assign the value +1 to any unknown element.

Step 2 Determine the unknown elements that are constrained by this assignment.

Step 3 Repeat steps 1 and 2 until there are no remaining unknown elements in P.

After the completion of part (4), the first P-matrix is determined. The set of independent elements consists of the q elements that have been assigned the value +1. It is important to point out that the choice of a set of independent elements is not unique. However, the number of independent elements is always unique.

The final segment of the procedure consists of three steps:

Step 1 Assign a set of values to the independent elements determined in part (4).

Step 2 Determine the remaining unknown elements in P.

Step 3 Repeat steps 1 and 2 until the remaining 2^q-1 distinct port orientation matrices are determined.

Example Consider the following conductance matrix.

$$G = \begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & 6 & 0 & -3 & 0 & 2 & 0 & 0 \\ 2 & 0 & 5 & 0 & 1 & 0 & 3 & 0 \\ 3 & -3 & 0 & 16 & 0 & -6 & 0 & -5 \\ 4 & 0 & 1 & 0 & 12 & 0 & 5 & 4 \\ 5 & 2 & 0 & -6 & 0 & 10 & 0 & 0 \\ 6 & 0 & 3 & 0 & 5 & 0 & 10 & 0 \\ 7 & 0 & 0 & -5 & 4 & 0 & 0 & 11 \end{array}$$

There are 13 zero elements in the upper triangular portion of G. The application of Theorem 1 results in the following entries in the matrix P:

$$P = \begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & 1 & x & -1 & x & 1 & x & x \\ 2 & x & 1 & x & 1 & x & 1 & x \\ 3 & -1 & x & 1 & x & -1 & x & -1 \\ 4 & x & 1 & x & 1 & x & 1 & 1 \\ 5 & 1 & x & -1 & x & 1 & x & x \\ 6 & x & 1 & x & 1 & x & 1 & x \\ 7 & x & x & -1 & 1 & x & x & 1 \end{array}$$

where the symbol "x" indicates that the corresponding element is still unknown. There does not exist a unique submatrix P_1 for any unknown element in P. Therefore, P is not unique. A set of independent elements is p_{17}, p_{14}, p_{12}. The set of P_1-submatrices used to determine the remaining unknown elements in P is shown in Table 1. The following is an example of the calculation of an unknown element.

Determination of element p_{57} ($p_{17} = p_{14} = p_{12} = +1$)

$$P_1 = \begin{array}{c|cccc} & 5 & 1 & 3 & 7 \\ \hline 5 & 1 & 1 & -1 & x \\ 1 & & 1 & -1 & 1 \\ 3 & & & 1 & -1 \\ 7 & & & & 1 \end{array} \qquad G_1 = \begin{array}{c|cccc} & 5 & 1 & 3 & 7 \\ \hline 5 & 10 & 2 & -6 & 0 \\ 1 & & 6 & -3 & 0 \\ 3 & & & 16 & -5 \\ 7 & & & & 11 \end{array}$$

Since $p_{51}\ p_{53} = p_{17}\ p_{37} = p_{13}$, part (1) of Theorem 7 is applied. The values of M_1, M_2, m_1, and m_2 are,

$$M_1 = 2$$
$$M_2 = 0$$
$$m_1 = 1$$
$$m_2 = 1$$

Since $|g_{13}| = 3 > M_1$ and $> M_2$ and since $m_1 = m_2$, $p_{57} = -\ p_{51}\ p_{17} = -1$. The first P-matrix is

$$P = \begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & 1 & 1 & -1 & 1 & 1 & -1 & 1 \\ 2 & 1 & 1 & -1 & 1 & -1 & 1 & 1 \\ 3 & -1 & -1 & 1 & -1 & -1 & 1 & -1 \\ 4 & 1 & 1 & -1 & 1 & -1 & 1 & 1 \\ 5 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\ 6 & -1 & 1 & 1 & 1 & 1 & 1 & -1 \\ 7 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \end{array}$$

The associated port-tree is shown in Figure 1. Since $q = 3$ there are $2^3 = 8$ port orientation matrices corresponding to G. Table 2 shows the dependence of the 10 remaining unknown elements in P on the three independent elements p_{17}, p_{14}, and p_{12}.

V. CONCLUSION

A method has been presented for determining all possible P-matrices (port tree-structures) from a given short-circuit admittance matrix Y(s). Some advantages of the technique are: (a) P-matrices are quickly determined from Y-matrices that have a small number of zeros. (b) The total number of P-matrices that correspond to a given Y-matrix is determined directly from any valid P-matrix. (c) The method is easily programmed for digital computation. A computer program ("POM") based on this technique has been written.

ACKNOWLEDGMENT

The authors are indebted to the National Science Foundation for their partial support of this work (NSF Grant GK-1044).

REFERENCES

1. F. T. Boesch and D. C. Youla, "Synthesis of N+1 Node Resistive N-Ports," IEEE Transactions on Circuit Theory, Vol. CT-12, pp. 515-520, December 1965.

2. C. C. Halkias, I. Cederbaum, and W. H. Kim, "Synthesis of Resistive N-Port Networks with N+1 Nodes," IRE Transactions on Circuit Theory, Vol. CT-9, pp. 69-73, March 1962.

3. P. P. Civalleri, "A Direct Procedure for the Synthesis of Resistive (n+1)-Poles," *IEE (London) Monograph No. 464E*, pp. 76-82, August 1961.

4. G. Biorci, "Sign Matrices and Realizability of Conductance Matrices," *IEE (London) Monograph No. 424E*, pp. 296-299, December 1960.

5. E. A. Guillemin, *Theory of Linear Physical Systems*, John Wiley and Sons, New York, New York, May 1963, pp. 80-90.

6. D. J. Schmidlin and F. J. Lupo, "The Determination of Port Orientation Matrices from Short-Circuit Admittance Parameter Matrices," *New York University Technical Report* (to be published).

TABLE 1

k	i	j	m	ELEMENT DETERMINED
5	1	3	7	p_{57}
3	1	7	4	p_{34}
4	1	7	5	p_{45}
2	1	4	3	p_{23}
2	1	4	5	p_{25}
2	3	4	7	p_{27}
6	2	4	7	p_{67}
3	2	7	6	p_{36}
1	2	3	6	p_{16}
5	1	2	6	p_{56}

TABLE 2

p_{12}	p_{14}	p_{17}	p_{16}	p_{23}	p_{25}	p_{27}	p_{34}	p_{36}	p_{45}	p_{56}	p_{57}	p_{67}
1	1	1	-1	-1	-1	1	-1	1	-1	1	-1	-1
1	1	-1	-1	1	-1	1	1	-1	-1	1	1	-1
1	-1	1	-1	-1	-1	-1	1	1	1	1	-1	1
1	-1	-1	-1	1	-1	-1	-1	-1	1	1	1	1
-1	1	1	1	1	1	-1	-1	-1	-1	-1	-1	1
-1	1	-1	1	-1	1	-1	1	1	-1	-1	1	1
-1	-1	1	1	1	1	1	1	-1	1	-1	-1	-1
-1	-1	-1	1	-1	1	1	-1	1	1	-1	1	-1

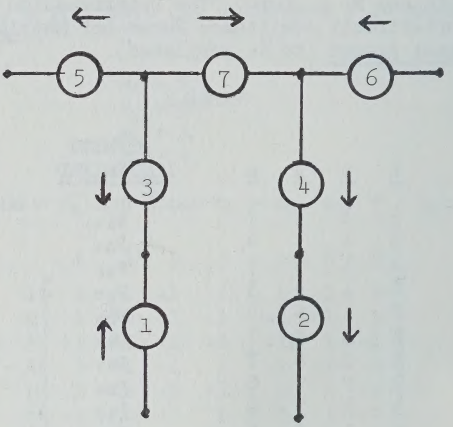


FIG. 1

ASSOCIATED PORT-TREE OF FIRST PORT ORIENTATION MATRIX IN EXAMPLE

PERFECT MATCHING FOR INFINITELY ITERATED LOSSLESS STRUCTURES (With Applications to Practical Network Matching)

R. A. Rohrer
Department of Electrical Engineering
University of California, Berkeley

ABSTRACT

Given a real, lumped, linear, time-invariant, strictly energy absorbing (realistically, strictly passive) one-port network characterized by the driving-point impedance function $Z(p)$; it is theoretically optimally matched by a real, lumped, linear, time-invariant, strictly energy absorbing one-port network characterized by the driving-point impedance function $\hat{Z}(p) = Z(-p)$. Aside from trivial examples involving memoryless networks, this ideal may only be achieved for the driving-point impedances of infinitely iterated lossless two-port networks, which become lossy in the limit. This observation leads to a technique of generation of matching network structures and to the design of matching networks of arbitrary complexity.

INTRODUCTION

A general matching problem for real, linear, time-invariant one-port networks is represented symbolically in Fig. 1:

> Find the real, linear, time-invariant one-port loading network $\hat{\mathcal{N}}$ which over the doubly-infinite frequency interval, $\omega \in (-\infty, \infty)$ absorbs the maximum net energy from the given real, linear, time-invariant network $\mathcal{N}$ for arbitrary excitation.

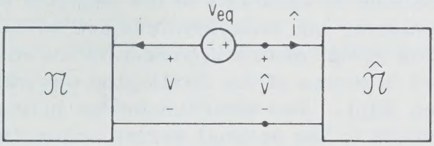


Fig. 1. The (Thévenin equivalent for the) general matching situation, which gives the basis on which the matching problem is enunciated and on which its solution is stated.

The solution of this matching problem is given in the following

Theorem [1, 2] : A given real, linear, time-invariant one-port network $\mathcal{N}$, which is (zero-state) characterized by the driving-point impedance function $Z(p)$, is optimally matched by the real, linear, time-invariant one-port network $\hat{\mathcal{N}}$, (zero-state) characterized by the driving-point impedance function

$$\hat{Z}(p) = Z(-p), \qquad p \neq \infty,$$

if it is strictly energy absorbing,

$$\underline{\mathrm{Re}}\,[Z(j\omega)] > 0 \quad \forall \quad \omega \in (-\infty, \infty),$$

and if the following assumptions hold:

(i) Both the given one-port network $\mathcal{N}$ and the matching one-port network $\hat{\mathcal{N}}$ are assumed to be initially relaxed (contain no stored energy) as $t \rightarrow -\infty$;

(ii) All signal functions in question are square-integrable on the doubly-infinite frequency-interval, $\omega \epsilon (-\infty, \infty)$, in particular

$$V_{eq} \epsilon L_2(-\infty, \infty) \Longrightarrow \int_{-\infty}^{\infty} |V_{eq}(j\omega)|^2 d\omega < \infty ;$$

(iii) Only the particular solution to the network equations arises for any excitation $V_{eq} \epsilon L_2(-\infty, \infty)$ considered.

The utility of this theorem is enhanced by the following two

Corollaries: (i) Strict passivity is a sufficient condition that a network be strictly energy absorbing; (ii) Consideration only of some (symmetric) subset $\Omega \subset (-\infty, \infty)$ of the doubly-infinite frequency interval $(\omega \epsilon \Omega \Rightarrow -\omega \epsilon \Omega)$ entails only that

$$\underline{\mathrm{Re}}\,[Z(j\omega)] > 0, \qquad \omega \epsilon \Omega,$$

and strict passivity may be replaced with passivity with strict passivity only over the frequency interval of interest [3].

In general if we are given a lumped, linear, time-invariant one-port Thévenin equivalent network, which is comprised of an arbitrary interconnection of resistances, inductances, capacitances, and transformers,† characterized by the impedance function $Z(p)$; a theoretical realization of the optimal matching network characterized by the impedance function $Z(-p)$ may be obtained from the original merely upon the replacement of all inductances and capacitances by elements of the negative equal value (the topology, resistance values, and transformers are to be the same as in the original network). The result of this approach to the matching problem is most easily envisioned in terms of the Darlington equivalent realization of the impedance function $Z(p)$. The interconnection indicated symbolically in Fig. 2a always reduces to the optimal series connection of two one-ohm resistances as shown in Fig. 2b; hence, the optimal theoretical match is frequency independent.

The question arises as to when, if ever, the ideal matching situation dictated by

$$Z(p) = Z(-p)$$

is physically realizable. Certainly, if $Z(p)$ is a real positive constant representing a resistance (or a frequency-independent resistance, ideal transformer network),

$$Z(p) \equiv R,$$

---

† Gyrators may be excluded from consideration since their presence would not be evidenced in a driving-point impedance function; a similar statement could be made regarding transformers, but subsequent concern with Darlington representations of driving-point impedance functions precludes this dismissal.

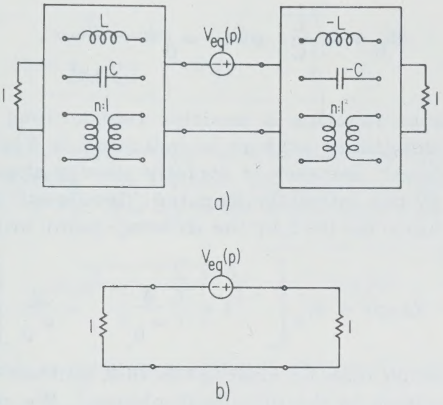


Fig. 2. An optimal match may be effected by considering the Darlington equivalent representation for a positive real impedance function Z(p) and replacing L → -L and C → -C ; the resulting interconnection (a) has the reduced equivalent (b).

---

then

$$\hat{Z}(p) = Z(-p) = R$$

yields the optimal matching network, realized by another resistance of the same value. However, if Z(p) is a (nonconstant) rational impedance function, Z(-p) is not physically realizable; and, even if it were, stability considerations would preclude the acceptance of such a matching situation. It would appear then that some reasonable approximation to Z(-p),

$$\hat{Z}(p) \approx Z(-p),$$

which is physically realizable and which renders the overall network asymptotically stable, is in order; the means for obtaining such an approximation is to be the subject of a succeeding section. However, before we abandon the quest for exact solutions we may examine the possibility of the employment of irrational driving-point impedance functions in matching situations.

PERFECT MATCHING

By way of motivation, we consider first the infinite ladder network comprised of a cascade of series inductances of value L and shunt capacitances of value C shown in Fig. 3 a. In the limit as this cascade of lossless elements approaches infinity, the driving-point impedance function becomes

$$Z(p) = R_0 \left[\sqrt{1 + \left(\frac{p}{\omega_0} \right)^2} + \frac{p}{\omega_0} \right],$$

where

$$R_0 \equiv \sqrt{\frac{L}{C}} \quad \text{and} \quad \omega_0 \equiv \frac{2}{\sqrt{LC}} .$$

This irrational impedance function is positive real so long as the branch cut is taken along the imaginary axis as is indicated in Fig. 3 b. This infinitely iterated "lossless" network is strictly energy absorbing[†] and may be optimally matched by the infinitely iterated "lossless" network shown in Fig. 3 c, which is characterized by the driving-point impedance function

$$\hat{Z}(p) = Z(-p) = R_0 \left[\sqrt{1 + \left(\frac{p}{\omega_0}\right)^2} - \frac{p}{\omega_0} \right] .$$

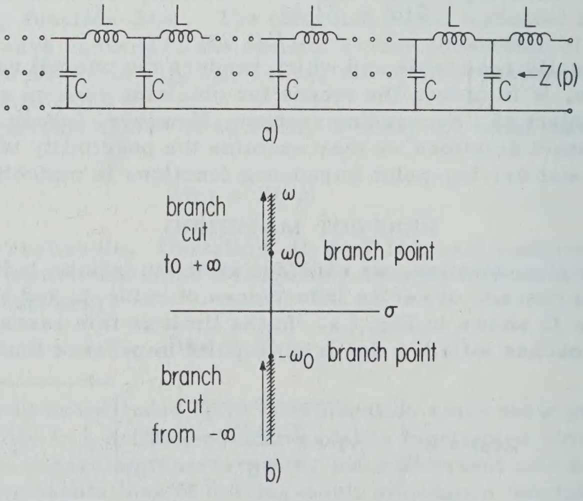
The reason that this match may be effected is that neither impedance function possesses singularities in the open half-planes. We recognize, moreover, that this frequency-dependent ideal matching situation has an effective bandwidth of

$$\omega_0 = \frac{2}{\sqrt{LC}}$$

since the voltage or current transfer function over k sections of the structure,

$$\frac{V_{m+k}(p)}{V_m(p)} = \frac{1}{\left[\sqrt{1+(p/\omega_0)^2} + p/\omega_0\right]^{2k+1}} = \frac{I_{m+k}(p)}{I_m(p)} ,$$

possesses the following limiting $(k \to \infty)$ sinusoidal frequency magnitude behavior:



a)

b)

[†] Consider only frequencies in the useful range, $|\omega| < |\omega_0|$.

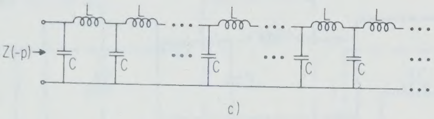


Fig. 3. An ideal match may be obtained for infinitely iterated lossless networks: The infinite LC ladder network (a) is characterized by the driving-point impedance function

$$Z(p) = \sqrt{\frac{L}{C}}\left[\sqrt{1 + \left(\frac{p\sqrt{LC}}{2}\right)^2} + \frac{p\sqrt{LC}}{2}\right],$$

which is strictly passive in the useful frequency range, $|\omega| < \frac{2}{\sqrt{LC}}$, so long as the branch cut is taken as indicated in (b); the infinite LC ladder network (c) is characterized by the ideal matching driving-point impedance function

$$Z(-p) = \sqrt{\frac{L}{C}}\left[\sqrt{1 + \left(\frac{p\sqrt{LC}}{2}\right)^2} - \frac{p\sqrt{LC}}{2}\right].$$

$$\left|\frac{V_{\infty}(j\omega)}{V_m(j\omega)}\right| = \begin{cases} 1, & |\omega| < \omega_0 \\ 0, & |\omega| > \omega_0 \end{cases}.$$

An immediate consequence of the foregoing example is that we recognize a means for assessing the restrictions on matching imposed by an intervening lossless two-port network. For the situation of Fig. 4 the match which can be obtained is limited by the effective bandwidth $\omega_0 = 2/\sqrt{LC}$, which is taken to be a figure of merit for the given lossless two-port network. We may generalize this approach to handle arbitrary intervening lossless two-port networks in matching situations. Given the infinitely iterated lossless, reciprocal two-port network shown in Fig. 5; the two designated driving-point impedance functions $Z(p)$ and $\hat{Z}(p)$ are given by

$$Z(p) = \frac{z_{11}(p) - z_{22}(p)}{2} + \sqrt{\left[\frac{z_{11}(p) - z_{22}(p)}{2}\right]^2 + [\Delta(p)]}$$

and

$$\hat{Z}(p) = \frac{z_{22}(p) - z_{11}(p)}{2} + \sqrt{\left[\frac{z_{22}(p) - z_{11}(p)}{2}\right]^2 + [\Delta(p)]},$$

where $\Delta(p)$ is the determinant of the symmetric two-by-two impedance matrix

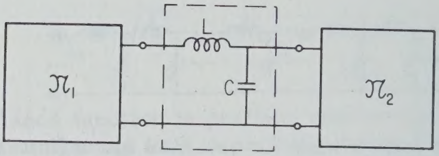


Fig. 4. In the optimal matching situation the effective bandwidth is $\omega_0 = \frac{2}{\sqrt{LC}}$; hence, if the networks $\mathcal{N}_1$ and $\mathcal{N}_2$ are restricted to be strictly passive, this effective bandwidth becomes a figure of merit for this matching situation.

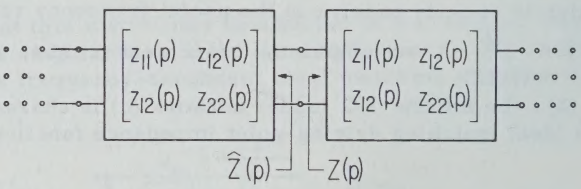


Fig. 5. For the infinitely iterated lossless reciprocal two-port network (two typical sections of which are shown), the ideal matching situation $\hat{Z}(p) = Z(-p)$ results.

$$Z(p) \equiv \begin{bmatrix} z_{11}(p) & z_{12}(p) \\ z_{12}(p) & z_{22}(p) \end{bmatrix} .$$

Since the network is lossless z_{11} and z_{22} are odd functions of p while Δ is even; hence,

$$\hat{Z}(p) = Z(-p)$$

and the ideal matching situation prevails. The driving-point impedance functions Z and Z are strictly positive real over their useful frequency range for arbitrary lossless two-port networks, except in the special pathological case for which

$$z_{11}(p) \equiv z_{22}(p) \equiv \pm\, z_{12}(p) .$$

For example, the infinite iteration of the lossless, reciprocal two-port network given in Fig. 6 a yields a short circuit, while the infinite iteration of that given in Fig. 6 b yields an open circuit; in neither case do we obtain a useful matching situation. The computation of effective bandwidth is most easily discussed in particular situations; we seek merely those frequency intervals for which

$$\left[\frac{z_{22}(j\omega) - z_{11}(j\omega)}{2}\right]^2 + [\Delta(j\omega)] \geqq 0 .$$

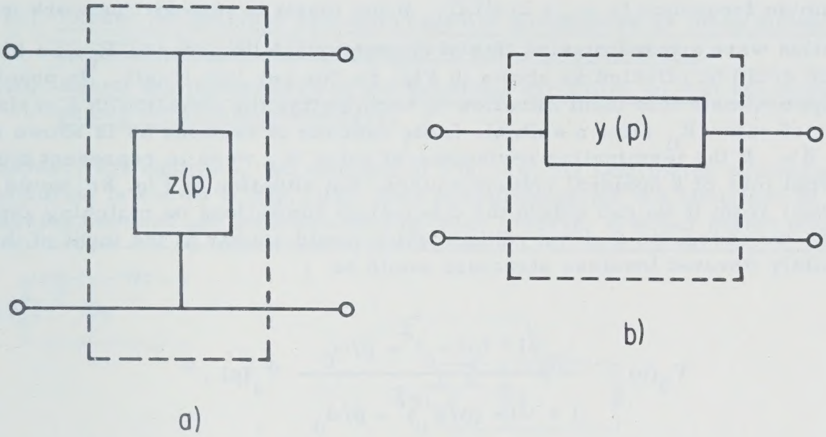


Fig. 6. Two pathological lossless, reciprocal two-port networks: (a) $z_{11}(p) \equiv z_{22}(p) \equiv z_{12}(p) \equiv z(p)$, which iterates to a shunt short-circuit; and (b) $y_{11}(p) \equiv y_{22}(p) \equiv -y_{12}(p) \equiv y(p)$, which iterates to a series open-circuit.

For example, the lossless reciprocal two-port network of Fig. 7 has the effective bandwidth

$$\omega_0 = \frac{2}{3},$$

which may be normalized further by the dc gain.

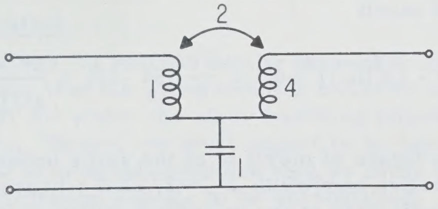


Fig. 7. A lossless, reciprocal two-port network the effective bandwidth of which is $\omega_0 = \frac{2}{3}$.

PRACTICAL IMPLICATIONS OF PERFECT MATCHING

Since it is unlikely that one would wish to employ infinitely iterated lossless networks in actual matching network design situations, we shall return to the motivational example to see how this theory may be bent to practice. The iterated network of Fig. 3 a is a lumped element approximation to an ideal transmission line of characteristic impedance $R_0 = \sqrt{L/C}$. However, the granularity error, introduced by the fact that the elements are lumped and their values are not infinitesimal, limits the effective maximum frequency to $\omega_0 = 2/\sqrt{LC}$. If the lossless two-port network in question were a transmission line of characteristic impedance R_0, an ideal match could be effected as shown in Fig. 8a for any line length. In practice we approximate this ideal situation by terminating the iteration in a resistance of value R_0 after a suitable finite cascade of sections as is shown in Fig. 8b. If the terminating resistance of value R_0 were to represent the internal loss of a nonideal voltage source, the situation of Fig. 8c would prevail; from it we can obtain the theoretical limitations on matching ability. First, we recognize that the voltage which would appear at the input of the infinitely iterated lossless structure would be

$$V_0(p) = \frac{\sqrt{1+(p/\omega_0)^2} - p/\omega_0}{1+\sqrt{1+(p/\omega_0)^2} - p/\omega_0}\, V_s(p) ,$$

while the current would be

$$I_0(p) = \frac{1/R_0}{1+\sqrt{1+(p/\omega_0)^2} - p/\omega_0}\, V_s(p) .$$

Upon consideration of the current-basis reflection coefficient [1]

$$\rho^I(p) = \frac{\sqrt{1+(p/\omega_0)^2} - p/\omega_0 - 1}{\sqrt{1+(p/\omega_0)^2} - p/\omega_0 + 1} ,$$

we obtain the figure of merit

$$\int_0^\infty [1 - |\rho^I(j\omega)|^2]\, d\omega = \frac{\omega_0}{8}(5\pi - 13) = \frac{5\pi - 13}{4\sqrt{LC}} .$$

We recognize that this figure of merit is of the same order of magnitude and comparable to the effective bandwidth $\omega_0 = 2/\sqrt{LC}$; hence, each of these measures is roughly indicative of the matching ability of the network.

There is little reason to pursue this line of reasoning further because the transfer function which results upon the termination of a finite cascade of lossless sections in a resistance is not nearly ideal. The magnitude of the voltage function,

$$\left|\frac{V_\infty(j\omega)}{V_s(j\omega)}\right| = \begin{cases} \dfrac{1}{\sqrt{1+\sqrt{1-(\omega/\omega_0)^2}}}, & |\omega| < \omega_0 \\ 0, & |\omega| > \omega_0 \end{cases} ,$$

(ii) Iterate the lossless two-port network so obtained as many times as practical (or desirable) and terminate in a resistance ;

(iii) Adjust the parameters of the resulting structure so that the most nearly ideal match emerges.

Consider, for example, the network of Fig. 10 a; its Darlington equivalent realization is shown in Fig. 10 b. Two possibilities for the structure of the matching network are indicated in Fig. 10 c; clearly, a better match entails a greater number of sections.

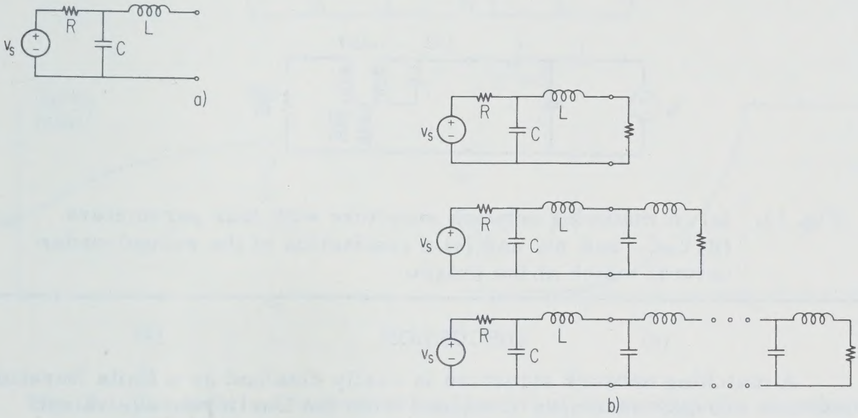


Fig. 9. A given network (a) and various degrees of approximate matching structures appropriate to the problem (b).

Passive Approximation

It is impractical to realize infinite networks; and it is, moreover, improbable, anyway, that the given network would be specified by an impedance function Z(p) for which the ideal matching impedance function Z(-p) would be realizable. Hence, we must expect to be compelled to resort to approximation, and it is at this juncture that we must become cognizant of the nature of the excitation signals to be encountered. For example, if the signals to be employed are comprised solely of low frequencies, Z(-p) need not be realized exactly -- a reasonable approximation to Z(-p) near $p = 0$ may suffice.

We obtain the natural match of order n at the frequency ω_0 by imposing the following specifications:

$$\frac{d^k}{dp^k}[\hat{Z}(p)]\bigg|_{p=j\omega_0} = \frac{d^k}{dp^k}[Z(-p)]\bigg|_{p=j\omega_0} \qquad (k = 0, 1, 2, \ldots, n) ;$$

or

$$\hat{Z}^{(k)}(j\omega_0) = (-1)^k Z^{(k)}(j\omega_0) \qquad (k = 0, 1, 2, \ldots, n).$$

lossless portion of the Darlington equivalent network yields a matching network structure with four parameters (R, L, C, and n) as shown in Fig. 11a, a third-order natural match at the origin is not physically realizable. Fig. 11b gives a realization of the second-order natural match at the origin.

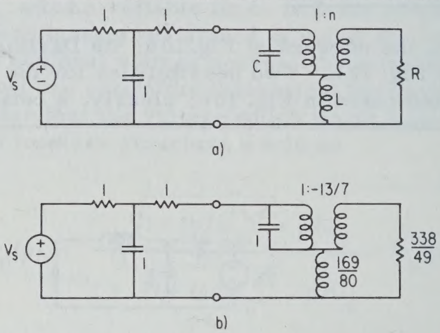


Fig. 11. (a) A matching network structure with four parameters (R, L, C, and n), and (b) a realization of the second-order natural match at the origin.

DISCUSSION

A matching network structure is easily obtained as a finite iteration of lossless two-port networks (obtained from the Darlington equivalent realization of the original) terminated in a resistance. While this pragmatic approach to the solution of the matching problem is attractive in its extreme simplicity, it perhaps raises as many questions as it resolves:

(i) Can any realizable approximation be achieved with the iterated structure?

(ii) Can network and frequency transformations be employed to generate a structure capable of matching in a frequency range not given by the immediate iteration of the lossless portion of the original Darlington equivalent network?

(iii) Can the approximation scheme be more closely controlled by the iterated structure so that realizability questions do not arise?

(iv) Can transformers and coupled coils be eliminated from the matching network in some optimal fashion.

The answers to these questions would certainly render this interesting matching scheme useful.

ACKNOWLEDGMENT

This paper is based on work which has been supported by a number of agencies: begun while the author was a National Academy of Sciences -- National Research Council Postdoctoral Fellow with the Department of Electrophysics, Polytechnic Institute Brooklyn (Farmingdale, N. Y.); it has continued at the Electronics Research Laboratory, University of Calif. (Berkeley, California) under NSF Grant GK-716, and most recently under

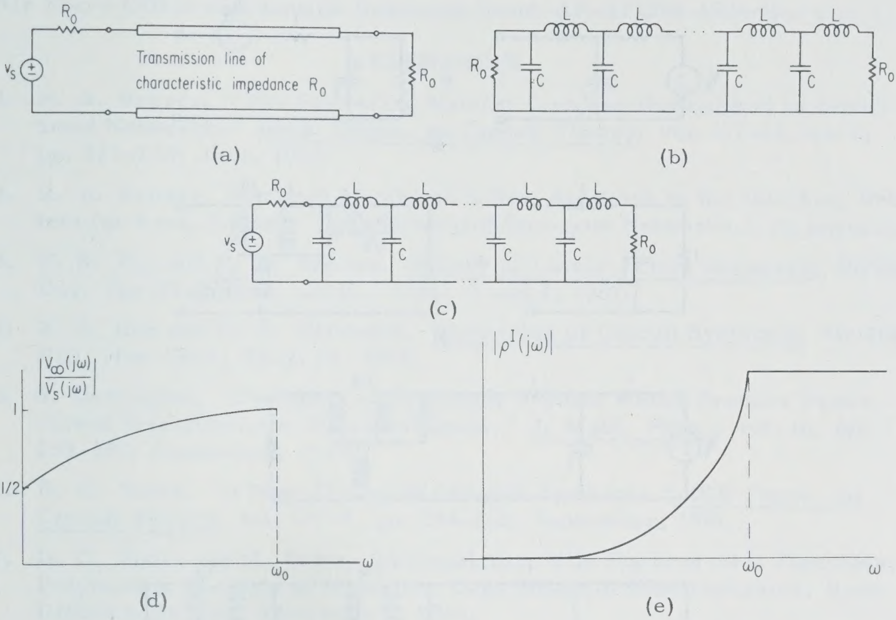

R_0
v_s
Transmission line of
characteristic impedance R_0
R_0
(a)
L
R_0
C
(b)
R_0
L
v_s
C
(c)
$\left|\frac{V_\infty(j\omega)}{V_s(j\omega)}\right|$
1
1/2
ω_0
ω
(d)
$|\rho^I(j\omega)|$
ω_0
ω
(e)

Fig. 8. The matched transmission line (a) of characteristic impedance R_0 and arbitrary length is the distributed network equivalent of the lumped network matching situation (c) which is the result of terminating a finite cascade of series L, shunt C sections in resistances of value $R_0 = \sqrt{\frac{L}{C}}$ as shown in (b). The limiting form of the voltage transfer characteristic is shown in (d), while the magnitude of the reflection coefficient for the limiting case is given in (e).

---

is plotted against frequency in Fig. 8d. The magnitude of the reflection coefficient $|\rho^I(j\omega)|$ is plotted against frequency in Fig. 8e for comparison. (It is recognized that the frequency dependence could be improved somewhat were m-derived [4] terminations to be employed; however, one can hardly expect a matching problem to come so conveniently equipped.) Nonetheless, the foregoing study remains significant because of the matching network structure which has emerged.

Structure Generation

A good practical match to the network of Fig. 9a is obtained with the finite structures shown in Fig. 9b upon the appropriate adjustment of the free parameters.

Extrapolating from the above, we may accept the following scheme for generating a matching network structure for a given lumped real, linear, time-invariant strictly passive one-port network:

(i) Find the Darlington equivalent realization for the given rational strictly positive real impedance function $Z(p)$ [5, 6];

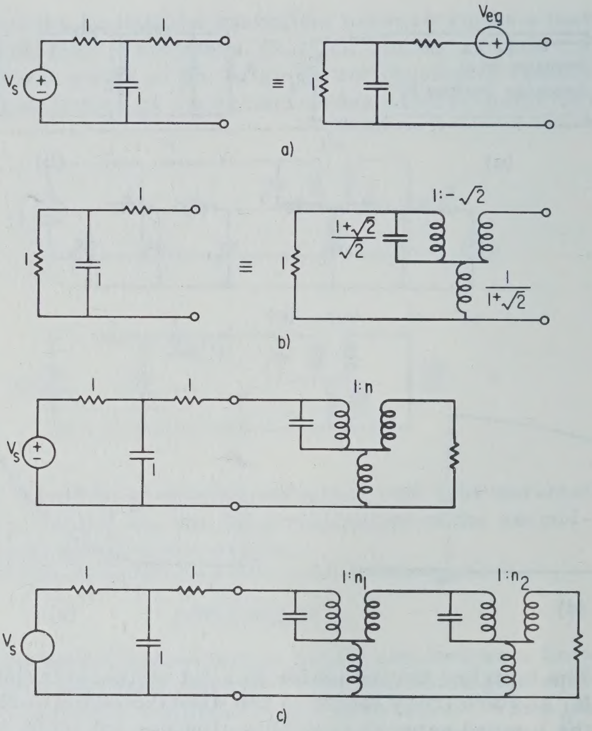


Fig. 10. To obtain a matching structure for the given network (a) find the Darlington equivalent of its Thévenin equivalent (b) and iterate the resulting lossless two-port network (the structures for one and two iterations are shown in (c)).

To render the scheme realistic we recognize that a natural match of order n at a finite nonzero sinusoidal frequency must be accompanied by a natural match of order n at the negative of that frequency. Presumably any number of (center) frequencies may be chosen. Obviously, this approximation scheme may be extended to nonsinusoidal frequencies; however, the lack of utility for such an approach dictates that it not be discussed here. In any event the question of realizability of the natural match becomes one of interpolation with positive real functions [7].

The main disadvantage connected with the application of the natural matching approximation is that the associated network structure is not specified by the numbers generated and must be intuited. However, since a reasonable structure may be obtained from iterated Darlington equivalent lossless two-port networks, we may expect that a matching network structure is always available and that the natural approximation merely dictates parameter values. A second drawback is that the realizability of the natural matching approximation is not ensured a priori. We recognize this to be but a minor consideration, however, when we realize how computationally simple it is to arrive at this point in the matching network design. Consider, for example, the network of Fig. 10 a; although one iteration of the

Air Force Office of Scientific Research Grant AF-AFOSR-1219-67.

REFERENCES

1. R. A. Rohrer, "The Scattering Matrix: Complex Normalized to n-port Load Networks," *IEEE Trans. on Circuit Theory*, vol. CT-12, no. 2, pp. 223-230; June, 1965.
2. R. A. Rohrer, "Optimal Matching: A New Approach to the Matching Problem for Real, Linear, Time-invariant One-port Networks," (to appear).
3. E. S. Kuh and R. A. Rohrer, *Theory of Linear Active Networks*, Holden-Day, San Francisco, Calif., Chap. 3 and 4; 1967.
4. E. S. Kuh and D. O. Pederson, *Principles of Circuit Synthesis*, McGraw-Hill, New York, Chap. 13; 1959.
5. S. Darlington, "Synthesis of Reactance 4-poles Which Produce Prescribed Insertion Loss Characteristics," *J. Math. Phys.*, vol. 18, pp. 257-353; September, 1939.
6. D. C. Youla, "A New Theory of Cascade Synthesis," *IRE Trans. on Circuit Theory*, vol. CT-8, pp. 244-260; September, 1961.
7. D. C. Youla and M. Saito, "Interpolation With Positive-real Functions," Polytechnic Institute of Brooklyn, Department of Electrophysics, Rept. PIBMRI-1353-66; November 7, 1966.

OPTIMUM FILTERING WITH A CLASS OF NONLINEAR SYSTEMS*

A. H. HADDAD
Coordinated Science Laboratory, University of Illinois
Urbana, Illinois

ABSTRACT

This paper considers the problem of nonlinear filtering and interpolation of random signals in the presence of noise. A special class of nonlinear systems is proposed which is composed of a linear system in parallel with a zero-memory nonlinearity preceded by a variable delay. The optimization problem is simplified if the inputs are restricted to be of the separable class. The optimum system is shown to be implemented in terms of the optimum linear filter and the optimum zero-memory filter. Furthermore, a suboptimal filtering scheme can also be constructed by adding a zero-memory nonlinearity to the optimum linear system which improves its performance.

INTRODUCTION

The nonlinear filtering problem for systems with random inputs is of importance in communication and control theory. Two basic approaches to the problem have been considered in the literature. The first is the classical approach which is based on Wiener's theory of filtering and is discussed in [1-4], among others. The second is based on the state variables representation and is considered in various papers, such as [5-7]. In both cases the problem involves mathematical and practical difficulties in the manipulation of the statistical properties of the nonlinearly processed inputs. Usually the processes involved are assumed to be Gaussian which in many cases facilitates the solution of the problem. However, the problem for non-Gaussian inputs is difficult in general, and there is no general approach which is convenient for all cases.

One of the methods suggested to simplify the optimization problem is based on the classification of nonlinear systems, and the restriction of the desired filter to a given class. In this manner the class of filters to be employed is adapted to the type of problem and to the properties of the inputs. Discussion of such classes may be found in [1-3]. Optimization of special subclasses of such filters are also considered in [8-10]. The restriction of the system to a given class of nonlinear filters usually simplifies the optimization and implementation procedures. The choice of the class of systems depends on the available statistical properties of the inputs among other factors. If only the first-order probability densities of the inputs are known, the class of filters which may be used is the zero-memory one. If the second-order densities of the inputs are known, class η_1 (introduced by Zadeh) may be used. However, it may be more convenient in some cases to use simpler subclasses to avoid difficulties in optimization.

---

*This work was supported by the Joint Services Electronics Program under Contract Army DAAB 07-67-C-0199.

The purpose of this paper is to suggest a simple class of nonlinear filters, composed of a linear system in parallel with a zero-memory nonlinearity (ZNL) preceded by a variable delay. It will be shown that this class of filters is very convenient for the filtering of a class of inputs whose second-order density satisfy the separability property [11-12]. The minimum-mean-squared error (MMSE) reconstruction and filtering of sampled data with such a class of filters is considered. The optimum filter is shown to be a combination of the optimum zero-memory nonlinear filter and the optimum linear filter. The simplification is in the fact that the second-order probability density of the input is not required, and the only equations which have to be solved involve the same properties required for the derivation of the optimum linear filter. The first step of the optimization results in a suboptimal filtering scheme which may be derived by adding a ZNL to the optimum linear system. Furthermore, the error improvement of the suboptimal scheme is almost the same as that of the optimum scheme for many practical cases. The error improvement depends also on the delay introduced in the nonlinear system, and the best choice of the delay is investigated and illustrated by an example.

PRELIMINARY DEFINITIONS

The class of filters proposed may be considered as a special case of Zadeh's class η_1, [1]. It is composed of a linear filter in parallel with a zero-memory nonlinearity preceded by a variable delay. Formally, the output of the system of such a class may be represented by the following expression

$$y(t) = \int_0^\infty h(\tau)x(t-\tau)\,d\tau + g[x(t-\ell)] \qquad (1)$$

where $x(t)$ is the input, $h(\tau)$ is the response of the linear part of the system, $g(x)$ is the ZNL, and ℓ is the delay. The system representation and the filtering problem are illustrated in Fig. 1. Here $s(t)$ represents the input corrupted by the additive noise $n(t)$. The desired output $y(t)$ is given by some linear (not necessarily realizable) operation on $s(t)$. The inputs $s(t)$ and $n(t)$ are assumed to be independent second-order stationary random processes having zero means, variances σ_s^2 and σ_n^2, and normalized correlation functions $\rho_s(\tau)$ and $\rho_n(\tau)$ respectively, i.e.,

$$\sigma_s^2\rho_s(\tau) = E\{s(t)\ s(t+\tau)\},\ \sigma_n^2\rho_n(\tau) = E\{n(t)\ n(t+\tau)\}. \qquad (2)$$

Furthermore, the inputs are assumed to be of the separable class [11], i.e. the conditional expectations satisfy the relationships

$$E\{s(t+\tau)|\ s(t)\} = \rho_s(\tau)\ s(t),\ E\{n(t+\tau)|\ n(t)\} = \rho_n(\tau)\ n(t) \quad (3)$$

The available input $x(t)$ therefore has variance $\sigma_x^2 = \sigma_s^2/\alpha$ and a normalized correlation function $\rho_x(\tau) = \alpha\rho_s(\tau) + (1-\alpha)\rho_n(\tau)$ where we have defined $\alpha = \sigma_s^2/(\sigma_n^2 + \sigma_s^2)$. The MMSE zero-memory filtering of $s(t)$ from $x(t)$ is given by the nonlinearity

$$g_s(x) = E\{s(t)|x(t) = x\}. \qquad (4)$$

The ZNL $g_s(x)$ is determined by the first order probability densities of the inputs and satisfies the relations

$$E\{s g_s(x)\} = E\{g_s^2(x)\} \underset{=}{\Delta} \beta\sigma_s^2, \quad E\{x g_s(x)\} = E\{s^2\} = \sigma_s^2 \tag{5}$$

where the constant $\beta \leq 1$ is defined by Eq. (5). The difference $(\beta-\alpha)$ represents the normalized improvement in the MMSE of the ZNL over the best linear attenuator.

The discrete input case will first be considered. The input $x(t)$ is sampled uniformly with sampling interval T, and the filter is assumed to operate on (N+1) samples. The filter is shown in Fig. 2, and its output may be expressed as

$$\hat{y}(t) = \alpha \sum_{i=0}^{N} a_i x_{k-i} + g(x_{k-\ell}) \tag{6}$$

where we have used the notation $x_k = x(kT)$ and where the constant α was introduced for convenience. The problem is the optimum reconstruction in the MMSE sense of $y(t)$ from the samples $(x_k, x_{k-1}, \ldots, x_{k-N})$. The desired output $y(t)$ may be expressed as

$$y(t) = \int_{-\infty}^{\infty} h_d(\tau)\, s(kT-\tau)\, d\tau \tag{7}$$

where the difference $(t-kT)$ is incorporated in $h_d(\tau)$. Suitable choices of $h_d(\tau)$ may distinguish between the problem of pure digital processing of digital inputs and the problem of reconstruction of continuous inputs sampled uniformly. As an example consider the special case $h_d(\tau) = \delta(\tau-\lambda T)$, i.e., $y(t) = s(kT-\lambda T)$. The factor λ illustrates the various types of problems: If $\lambda = 0,1,\ldots,N$, a pure filtering problem results. For λ positive non-integer the problem is one of interpolation or reconstruction and filtering. Negative values of λ yield the problem of filtering with prediction.

To simplify the expressions, the correlation between $y(t)$ and $s(kT-\ell T)$ will be denoted by

$$\rho_{sy}(\ell) = \frac{1}{\sigma_s^2} E\{y(t)\, s(kT-\ell T)\} = \int_{-\infty}^{\infty} h_d(\tau)\, \rho_s(\ell T-\tau)\, d\tau. \tag{8}$$

The following (N+1)-dimensional column vectors will also be introduced

$$\underline{a}' = [a_0, a_1, \ldots, a_N] \; ; \quad \underline{x}' = [x_k, x_{k-1}, \ldots, x_{k-N}]$$

$$\underline{r}'_{sy} = [\rho_{sy}(0), \rho_{sy}(1), \ldots, \rho_{sy}(N)] \; ; \quad \underline{q} = [\underline{r}_s(\ell) - \underline{r}_n(\ell)] \tag{9}$$

$$\underline{r}'_s(\ell) = [\rho_s(-\ell T), \rho_s(T-\ell T), \ldots, \rho_s(NT-\ell T)], \text{ similarly for } \underline{r}_n(\ell)$$

where the prime is used to denote transposed vectors. The input correlation matrix will be denoted by $R = E\{\underline{x}\cdot\underline{x}'\}/\sigma_x^2$ with entries $\{\rho_x(iT-jT)\}$, $i,j = 0,1,\dots,N$.

DERIVATION OF THE OPTIMAL FILTER

With the use of the notations of the previous section, the problem may be formulated as follows: It is desired to derive the nonlinearity $g(x)$ and the vector $\underline{a}$ which minimize the MSE as given by the expression

$$\overline{e^2} = E\{[y(t) - \hat{y}(t)]^2\} = E\{[\int_{-\infty}^{\infty} h_d(\tau)\, s(kT-\tau)\, d\tau - \alpha\underline{a}'\cdot\underline{x} - g(x_{k-\ell})]^2\}. \qquad (10)$$

The separability property of Eq. (3) may be used to derive the relations

$$E\{s(t+\tau)\,|\,x(t) = x\} = \rho_s(\tau) E\{s(t)\,|\,x(t) = x\} = \rho_s(\tau) g_s(x)$$

$$E\{n(t+\tau)\,|\,x(t) = x\} = \rho_n(\tau) E\{n|x\} = \rho_n(\tau)[x - g_s(x)]. \qquad (11)$$

The substitution of Eq. (11) in Eq. (10) results in a simplified expression for the MSE as follows

$$\overline{e^2} = E\{[y(t) - \alpha\underline{a}'\cdot\underline{x}]^2\} + E\{[g(x)]^2\} - 2E\{g_s(x)g(x)\}[\rho_{sy}(\ell) - \alpha\underline{a}'\cdot\underline{q}]$$
$$+ 2\alpha\, E\{xg(x)\}\underline{a}'\cdot\underline{r}_n(\ell). \qquad (12)$$

The minimization may be performed in two steps. First with respect to the nonlinearity $g(x)$ while the vector $\underline{a}$ is held fixed. Then the resulting MSE is minimized with respect to $\underline{a}$. The first step results in the following expression for the nonlinearity

$$g(x)\big|_{\underline{a}} = g_s(x)[\rho_{sy}(\ell) - \alpha\underline{a}'\cdot\underline{q}] - \alpha x\, \underline{a}'\cdot\underline{r}_n. \qquad (13)$$

It can easily be shown that a change in the linear factor in $g(x)$ and in the ℓ-th component of the vector $\underline{a}$ does not change the output or the MSE. Therefore we may modify the nonlinearity $g(x)$ to be given as

$$g(x) = A[g_s(x) - \alpha x] \qquad (14)$$

$$A = [\rho_{sy}(\ell) - \alpha\underline{a}'\cdot\underline{q}]. \qquad (15)$$

The resulting MSE may be obtained by substituting Eq. (14) in Eq. (12) to yield

$$\overline{e^2}\big|_{\underline{a}} = E\{[y(t) - \alpha\underline{a}'\cdot\underline{x}]^2\} - E\{[g(x)]^2\} =$$
$$= [E\{y^2(t)\} + \sigma_s^2(\alpha\underline{a}'\cdot R\cdot\underline{a} - 2\alpha\underline{a}'\cdot\underline{r}_{sy})] - \sigma_s^2(\beta-\alpha)A^2 \qquad (16)$$

where we have used the notations of Eq. (9) and the following relation

$$E\{g_s(x)[g_s(x) - \alpha x]\} = E\{[g_s(x) - \alpha x]^2\} = (\beta-\alpha)\sigma_s^2 .$$

The expression of the MSE after this first step of optimization suggests the use of two different filtering schemes as far as the linear filter (represented by the vector $\underline{a}$) is concerned.

Scheme I. The first scheme is obtained if we choose as the linear part, the optimum linear system, $\underline{a}_o$, as if the nonlinearity g(x) were not present. Such a choice implies that only the first term in Eq. (16) is minimized with respect to $\underline{a}$, since this term represents the MSE of the linear filtering scheme. Therefore, this system will be a suboptimal one, even though it is superior to the linear system alone as seen from Eq. (16). The vector $\underline{a}_o$ in this case will be given by

$$\underline{a}_o = R^{-1}\cdot\underline{r}_{sy} . \tag{17}$$

The corresponding linear MMSE is given by the expression

$$\epsilon_L = \frac{\overline{e_L^2}}{\sigma_s^2} = \frac{1}{\sigma_s^2} E\{[y(t) - \alpha\underline{a}_o'\cdot\underline{x}]^2\} = \frac{E\{y^2(t)\}}{\sigma_s^2} - \alpha\underline{a}_o'\cdot\underline{r}_{sy} . \tag{18}$$

The nonlinearity $g_o(x)$ to be added to the optimum linear system $\underline{a}_o$ to derive the suboptimal scheme is given by Eq. (14) after the substitution of $\underline{a}_o$ for $\underline{a}$, i.e.

$$\begin{aligned} g_o(x) &= A_o[g_s(x) - \alpha x] \\ A_o &= [\rho_s(\ell) - \alpha\underline{a}_o'\cdot\underline{q}] . \end{aligned} \tag{19}$$

The error improvement Δ_o of the suboptimal scheme over the linear optimal one may be derived from Eq. (16) and is expressed by

$$\Delta_o = \epsilon_L - \epsilon_o = \frac{\overline{e_L^2} - \overline{e_o^2}}{\sigma_s^2} = \frac{1}{\sigma_s^2} E\{[g_o(x)]^2\} = (\beta-\alpha)A_o^2 . \tag{20}$$

It is seen that the error improvement is proportional to the improvement, $(\beta-\alpha)$, of the zero-memory nonlinear filtering by the factor A_o^2 .

Scheme II. For the second scheme the vector $\underline{a}$ is chosen so as to minimize the MSE of Eq. (16). The minimization procedure results in the following set of linear equations for the optimum $\underline{a}$:

$$R\cdot\underline{a} = \underline{r}_{sy} - (\beta-\alpha)[\rho_{sy}(\ell) - \alpha\underline{a}'\cdot\underline{q}]\underline{q} \tag{21}$$

which may be solved for $\underline{a}$ to yield

$$\underline{a} = \underline{a}_o - (\beta-\alpha)A\,R^{-1}\cdot\underline{q} . \tag{22}$$

To obtain an explicit expression for the factor A, both sides of Eq. (22) are multiplied from the left by the vector $\alpha\underline{q}'$ and the resulting expression is

$$A = A_o/D, \quad D = 1-\alpha(\beta-\alpha)\underline{q}'\cdot R^{-1}\cdot\underline{q}\ , \quad 0<D\leq 1\ . \tag{23}$$

The improvement in the MMSE of the optimum nonlinear scheme over the optimum linear scheme will therefore be given by

$$\begin{aligned}\Delta = \epsilon_L - \epsilon_{min} &= \frac{\overline{e_L^2} - \overline{e_{min}^2}}{\sigma_s^2} = 2\alpha\underline{a}'\cdot\underline{r}_{sy} - \alpha\underline{a}'\cdot R\cdot\underline{a} + (\beta-\alpha)A^2 - \alpha\underline{a}_o'\cdot\underline{r}_{sy}\\ &= \alpha(\underline{a}'-\underline{a}_o')\cdot\underline{r}_{sy} + \alpha\underline{a}'(\underline{r}_{sy}-R\underline{a}) + (\beta-\alpha)A^2\\ &= (\beta-\alpha)AA_o = (\beta-\alpha)\frac{A_o^2}{D} = \frac{\Delta_o}{D}\end{aligned} \tag{24}$$

where use was made of Eqs. (16-18) and (20-23). Similarly it is seen from Eqs. (14) and (23) that the nonlinearity for the optimum scheme is given by $g_o(x)/D$.

To summarize: two nonlinear filtering schemes were derived. The first is constructed by adding the nonlinearity $g_o(x)$ of Eq. (19) to the optimum linear scheme $\underline{a}_o$ given by Eq. (17). The improvement in the error is given by Eq. (20). The second scheme is derived by changing both the ZNL and the linear filter of the suboptimal scheme. The vector $(\underline{a}-\underline{a}_o)$ of Eq. (22) represents the change to be made in the linear system. The nonlinearity is to be modified by the factor 1/D which is the same factor which modifies the error improvement. The convenience of both schemes lies in the fact that the optimum zero-memory filter and the optimum linear filter are sufficient to construct the filtering schemes. In the derivation of the change in the linear filter there is no need to invert a new matrix other than R, which has to be inverted to derive the optimum linear filter anyway. Furthermore, if the factor D is not significantly different from unity, it would be more practical to use the suboptimal scheme which is very easy to implement.

THE DEPENDENCE ON THE DELAY

So far the delay of ℓ samples was held fixed. However, the error improvement Δ_o depends on the delay, and therefore ℓ should be chosen so as to obtain the maximum improvement. It is not possible to derive analytic expressions for the best delay, especially in the discrete case, since it may take only the values (0,1,...,N). The dependence on the delay is exhibited through the factor A_o^2, which is given by Eq. (19) in which $\underline{q}$ also is a function of the delay. It would seen that the best delay should be chosen so as to maximize the correlation between y(t) and s(kT-ℓT), i.e., to maximize $|\rho_{sy}(\ell)|$. However, this statement is not true in general, and may be verified only for two special cases.

(1) The input s(t) has low power, i.e. $\alpha << \max_{0\leq\ell\leq N}|\rho_{sy}(\ell)|$.

(2) The signal and noise have the same normalized autocorrelation function, i.e., $\rho_s(\tau) = \rho_n(\tau)$.

In this case $\underline{q} = 0$, and therefore the error improvements are given by $\Delta = \Delta_o = [\rho_{sy}(\ell)]^2(\beta-\alpha)$, which verifies the statement.

Another case for which the best choice of delay can be found, is the problem of pure filtering, i.e., $y(t) = s(kT-\lambda T)$, λ is positive integer. It would seem that the best delay in this case is $\ell=\lambda$, which means that the nonlinearity is to be employed to the sample of $x(t)$ corresponding to the desired sample of $s(t)$. Again, this statement may be verified for the following cases:

(1) The noise has low power, i.e., $(1-\alpha) \ll 1$. In this case $A_o \approx \rho_n(\ell T-\lambda T)$.

(2) The linear part of the system is unrealizable with large memory. It can be shown under this assumption that $A_o = R_{ee}(\ell T-\lambda T)/\alpha\sigma_n^2$, where $R_{ee}(\tau)$ is the autocorrelation function of the error of the optimum linear filter. Therefore, for both cases the statement is verified since any autocorrelation function satisfies $|\rho(\ell T-\lambda T)| \leq \rho(0) = 1$.

In all other cases, it might be a reasonable assumption to try values of ℓ in the neighborhood of the delay determined by the special cases discussed above. For large memory N, z-transform methods and spectral factorization techniques may be used. However, these approaches will not be discussed here.

EXTENSION TO THE CONTINUOUS CASE

The solution for the continuous case is quite straightforward, resulting in the substitution of the linear equations by integral equations. In this case the filter output is given by

$$\hat{y}(t) = \alpha\int_0^b h(\tau)x(t-\tau)\,d\tau + g[x(\tau-\ell)] \tag{25}$$

and the desired output is given by

$$y(t) = \int_{-\infty}^{\infty} h_d(\tau)s(t-\tau)\,d\tau. \tag{26}$$

The optimum linear filter $h_o(\tau)$ is given by the solution of the Wiener-Hopf integral equation

$$\int_0^b h_o(\eta)\rho_x(\tau-\eta)\,d\eta = \int_{-\infty}^{\infty} h_d(\eta)\rho_s(\tau-\eta)\,d\eta, \quad 0 \leq \tau \leq b\ . \tag{27}$$

The suboptimal scheme is derived by adding the nonlinearity $g_o(x)$ as in Eq. (19) in parallel to the optimum linear filter, where A_o is given in this case by

$$A_o = \int_0^b h_o(\tau)\rho_n(\ell-\tau)\,d\tau\ . \tag{28}$$

The improvement in the error is given by the expression of Eq. (20). The optimum nonlinear scheme is derived by subtracting $\Delta h(\tau)$ from $h_o(\tau)$, where

$$\Delta h(\tau) = (\beta-\alpha)Au(\tau). \tag{29}$$

Here $u(\tau)$ is the solution of the integral equation

$$\int_0^b u(\eta)\rho_x(\tau-\eta)\,d\eta = [\rho_s(\tau-\ell) - \rho_n(\tau-\ell)], \quad 0 \leq \tau \leq b \tag{30}$$

which has the same kernel as the equation for the optimal linear filter. The constant A, the error improvement Δ, and the nonlinearity $g(x)$ are modified by the factor $1/D$ as for the discrete case, where

$$D = 1-\alpha(\beta-\alpha)\int_0^b u(\tau)[\rho_s(\tau-\ell) - \rho_n(\tau-\ell)]\,d\tau. \tag{31}$$

In this case too, for large memory spectral factorization techniques may be used, and the expressions for the filters and errors may be simplified. A numerical example will next be considered.

NUMERICAL EXAMPLE

An example for the continuous case with infinite memory is considered. The correlation functions are chosen to be

$$\rho_s(\tau) = \frac{1}{28}(33e^{-10|\tau|} - 5e^{-2|\tau|}), \quad \rho_n(\tau) = e^{-|\tau|}.$$

The problem of estimating $s(t+\lambda)$ is considered, where λ is chosen such that $\rho_s(\lambda)=0$. This choice is made so that $\ell_{max} \neq 0$, where ℓ_{max} is defined by

$$\max_{0 \leq \ell < \infty} |\rho_s(\lambda+\ell)| = |\rho_s(\lambda+\ell_{max})|.$$

In this case, it is found that $\ell_{max} = 0.201$. The improvement factor A_o as a function of the delay ℓ is derived, and the best delay ℓ_o is determined. The results are drawn in Fig. 3, which shows the best delay ℓ_o and corresponding improvement factor A_o as a function of the noise level $(1-\alpha)$. It is verified that for quite a large region of α, the best choice of the delay ℓ_o is not significantly different from ℓ_{max}. Furthermore it was also verified that for $\ell_{max} \leq \ell \leq \ell_o$ the change in A_o did not exceed 10%, so that choosing the delay as ℓ_{max} could be quite a reasonable assumption. The error improvement further depends on the difference $(\beta-\alpha)$ which is dependent on the first-order probability densities of the input. The factor D was found to be very close to unity, so that the difference between the optimal and suboptimal schemes is negligible.

CONCLUSION

A special class of nonlinear systems is applied to the optimum filtering and reconstruction of random inputs of the seperable class. It was shown that a suboptimal filtering scheme can be constructed from the optimum linear system and the optimum zero-memory nonlinear system. The error improvement is found to be proportional to the improvement of the optimum zero-memory nonlinear filtering system. An optimum nonlinear scheme of the same class can be constructed by modifying both the linear and the zero-memory nonlinear parts of the suboptimal system. The solution requires the inverse of the same matrix or the same kernel used to derive the optimum linear filter.

Further generalization of the class of filters and the class of inputs to be filtered may be possible. The inclusion of feedback or other system components could also be investigated. More definite procedures for deriving the best delay in the system may be worth considering.

REFERENCES

1. L. A. Zadeh, "Optimum Nonlinear Filters," J. of Applied Physics, Vol. 24, pp. 396-404, April 1953.

2. L. A. Zadeh, "On the Representation of Nonlinear Operatory," IRE Wescon Convention Record, Pt. 2, pp. 105-113, 1957.

3. J. K. Lubbock, "Nonlinear Filters," Progress in Radio Science 1960-1963, Vol. 6, pp. 127-166, 1965.

4. A. V. Balakrishnan, "A General Theory of Nonlinear Estimation Problems in Control Systems," J. of Math. Anal. and Appl., Vol. 8, pp. 4-30, February 1964.

5. W. M. Woham, "Some Applications of Stochastic Differential Equations to Optimal Nonlinear Filterings," TR 64-3, R.I.A.S., February 1964.

6. H. J. Kushner, "On the Differential Equations Satisfied by Conditional Probability Densities of Markov Processes, with Applications," TR 64-4, R.I.A.S., March 1964.

7. R. S. Bucy, "Nonlinear Filterings," IEEE Trans. on Automatic Control, Vol. AC-10, p. 198, April 1965.

8. J. K. Lubbock, "The Optimization of a Class of Nonlinear Filters," Proc. IEE (London), Pt. C, Vol. 107, pp. 60-74, March 1960.

9. N. K. Sinha, "Optimum Nonlinear Filters for Random Signals," IRE Convention Record, Vol. 10, Pt. 2, pp. 89-91, March 1962.

10. M. Lal, "A Note on Optimization of a Class of Nonlinear Systems," IEEE Trans. on Automatic Control, Vol. AC-11, pp. 137-138, January 1966.

11. A. H. Nuttall, "Theory and Application of the Separable Class of Random Processes," TR 343, Research Lab. of Electronics, M.I.T., May 1958.

12. J. L. Brown, "On a Cross-Correlation Property for Stationary Random Processes," IRE Trans. on Information Theory, Vol. IT-3, pp. 28-31, March 1957.

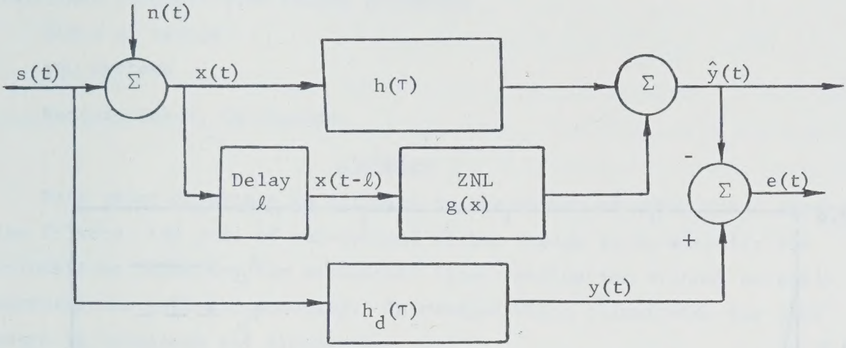

n(t)
s(t)
Σ
x(t)
h(τ)
Σ
ŷ(t)
Delay
ℓ
x(t-ℓ)
ZNL
g(x)
-
Σ
e(t)
+
y(t)
h_d(τ)

Fig. 1. The filtering problem.

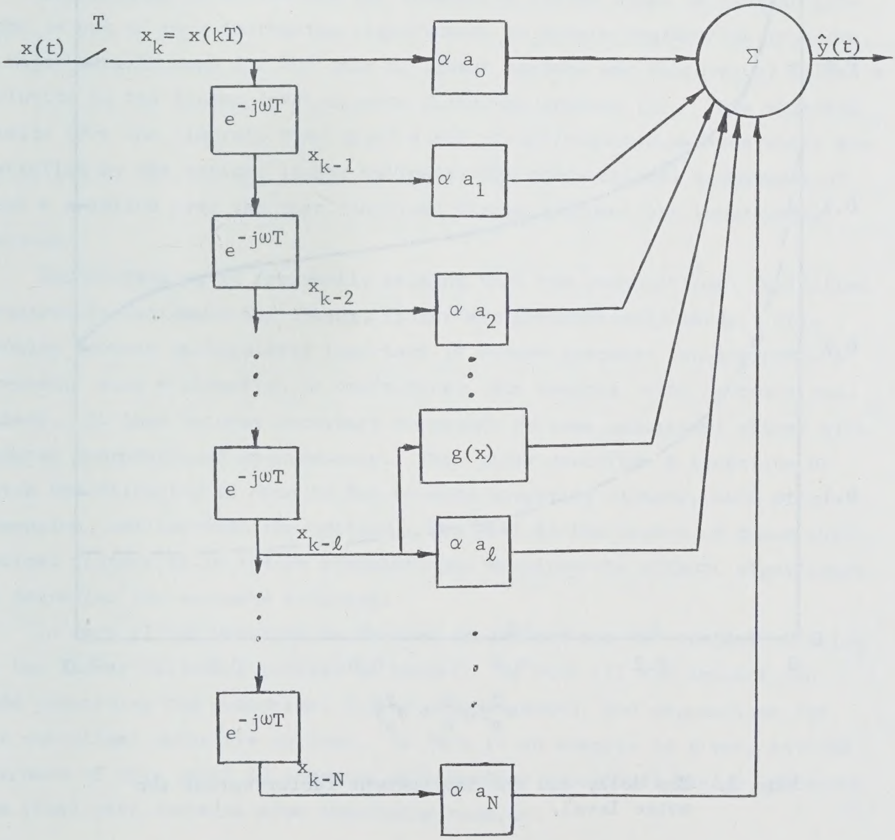

T
x(t)
x_k = x(kT)
α a_o
Σ
ŷ(t)
e^{-jωT}
x_{k-1}
α a_1
e^{-jωT}
x_{k-2}
α a_2
g(x)
e^{-jωT}
x_{k-ℓ}
α a_ℓ
e^{-jωT}
x_{k-N}
α a_N

Fig. 2. The representation of the nonlinear system for sampled data.

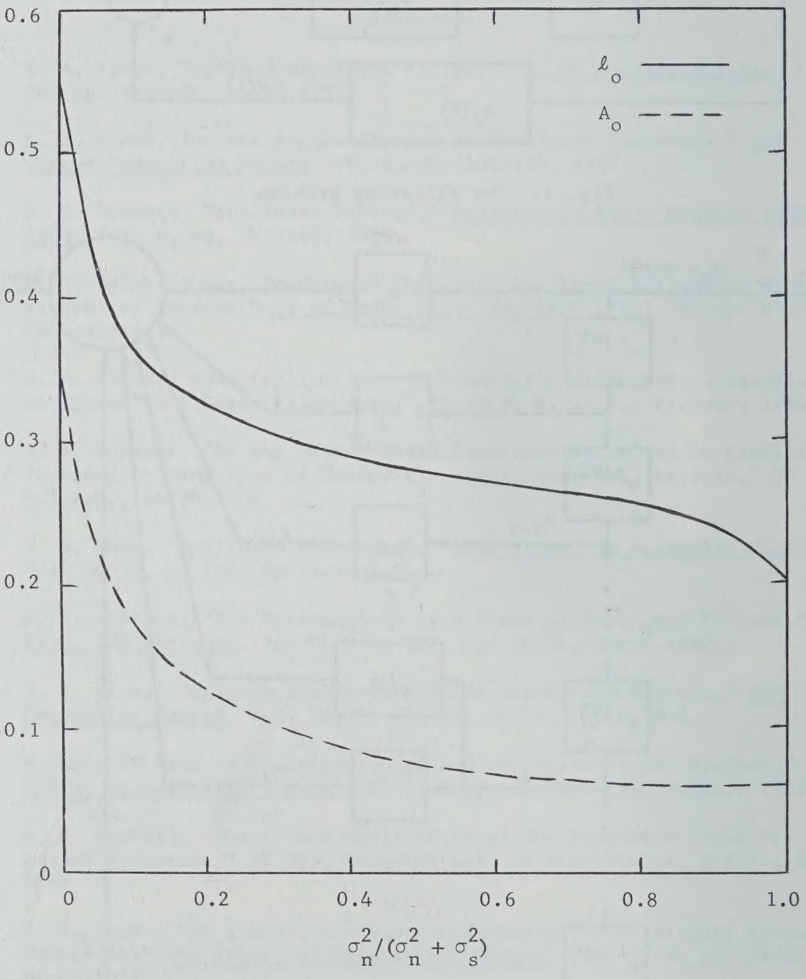


Fig. 3. The delay and the improvement factor versus the noise level.

SUBOPTIMAL DISCRETE TIME KALMAN FILTERING

DAVID M. LAYTON

TRW Systems

Redondo Beach, California

ABSTRACT

This paper describes an approach to the design of sub-optimal discrete time filters; the goal of sub-optimal filter design is to simplify the computations necessary for sequential linear estimation without severely degrading the estimate accuracy. An example which illustrates the procedure is presented and discussed.

I. INTRODUCTION

The problem of estimating the present or future state of dynamic systems is one of ever increasing significance in modern engineering problems. A major breakthrough for the case of linear systems was achieved by Kalman's solution to the linear least-squares filtering problem (1). This approach yields (for the discrete time case) a set of difference equations which are satisfied by the optimal linear estimate; the computational advantages of such a solution over the more classical Wiener approach are immediately obvious.

Unfortunately, it frequently happens that the computational facilities required to implement the optimal filter are prohibitively large. This problem becomes particularly important if severe computer constraints are present; such a situation is encountered, for example, with on-board computers. It then becomes necessary to resort to some suboptimal scheme with reduced computational requirements. This paper describes a technique by which the filtering is done in two or more uncoupled filters, each with dimension smaller than the optimum. The goal in the design of these suboptimal filters is to reduce computational requirements without significantly degrading the estimate accuracy.

In Part II the notation to be used is defined and the optimal solution to the linear filtering problem is stated. In Part III the assumptions made concerning the suboptimal filter are presented, and expressions for the suboptimal gains are derived. In Part IV an example is given, and the approach of this paper is compared with another suboptimal design procedure. The final part contains some concluding remarks.

II. NOTATION AND STATEMENT OF FILTERING PROBLEM

The system whose state is to be estimated is assumed to be described by a linear difference equation which has the form:

$$\begin{aligned} \underline{x}(t_{k+1}) &= \phi(t_k)\ \underline{x}(t_k) + \underline{z}(t_k) \\ \underline{y}(t_k) &= M(t_k)\ \underline{x}(t_k) + \underline{w}(t_k) \end{aligned} \tag{1}$$

$\underline{x}$ and $\underline{z}$ are n_x component random vectors; $\underline{y}$ and $\underline{w}$ are n_y component vectors; $\phi(t_k)$ and $M(t_k)$ are matrices of dimension $n_x \times n_x$ and $n_y \times n_x$, respectively; $\underline{z}$ and $\underline{w}$ are jointly Gaussian with moments:

$$E\left\{\underline{w}(t_k)\right\} = 0$$

$$E\left\{\underline{z}(t_k)\right\} = 0$$

$$E\left\{\underline{w}(t_k)\ \underline{z}^t(t_k)\right\} = 0$$

$$E\left\{\underline{w}(t_k)\ \underline{w}^t(t_j)\right\} = W(t_k)\ \delta_{kj}$$

$$E\left\{\underline{z}(t_k)\ \underline{z}^t(t_j)\right\} = Z(t_k)\ \delta_{kj}$$

Henceforward, explicit dependence of quantities on time will be omitted, unless confusion should arise.

The statement of the linear filtering problem is then the following: given the sequence of observations

$$\underline{y}(t_o),\ \ldots,\ \underline{y}(t_k),$$

find an estimate $\hat{x}(t_k|k) = \sum_{i=0}^{k} D(i) \times \underline{y}(t_k)$, (where the $D(i)$ are matrices of dimension $n_x \times n_y$), so that the expected value of the sum of the squares of the components of the error vector

$$\underline{\epsilon}(t_k|k) = \underline{x}(t_k) - \hat{\underline{x}}(t_k|k)$$

is minimized.

The solution to the problem is obtained as follows (Reference (2)): If we let $Y(k)$ be the linear space of random vectors of the form:

$$\sum_{i=0}^{k} B(i)\ \underline{y}(t_i)$$

(where $B(i)$ are arbitrary, real matrices of dimension $n_x \times n_y$), and if we norm the space of n_x component random vectors with the scalar product norm:

$$||\underline{x}|| = \ < \underline{x},\ \underline{x} > \ = \left\{E\left\{\sum_{i=1}^{n_x} x_i^2\right\}\right\}^{1/2},$$

then the desired optimal estimate is the orthogonal projection $\hat{\underline{x}}(t_k|k)$ of $\underline{x}(t_k)$ on the space $Y(k)$. This is computed with the following equations:

$$\hat{\underline{x}}(t_k|k) = \hat{\underline{x}}(t_k|k\text{-}1) + A^*\left[\underline{y}(t_k) - M\ \hat{\underline{x}}(t_k|k\text{-}1)\right] \tag{2}$$

where

$$\hat{\underline{x}}(t_k|k\text{-}1) = \phi\ \hat{\underline{x}}(t_{k-1}|k\text{-}1) \tag{3}$$

and

$$A^* = JM^t\left[MJM^t + W\right]^{-1} \tag{4}$$

The error covariance matrices

$$J(t_k) = E\left\{\underline{\epsilon}(t_k|k-1)\ \underline{\epsilon}^t(t_k|k-1)\right\} \text{ and } J^+(t_k) = E\left\{\underline{\epsilon}(t_k|k)\ \epsilon^t(t_k|k)\right\}$$

are given respectively by (5) and (6):

$$J(t_k) = \emptyset(t_{k-1})\ J^+(t_{k-1})\ \emptyset^t(t_{k-1}) + Z(t_{k-1}) \tag{5}$$

$$J^+(t_k) = \left[I - A^* M\right] J(t_k) \tag{6}$$

If some gain A other than the optimal A^* is used in Equation (2), the expression for the error covariance matrix becomes:

$$J^+(t_k) = \left[I - AM\right] J(t_k) \left[I - AM\right]^t + AWA^t \tag{7}$$

We will proceed now to a discussion of the suboptimal filter.

III. SUBOPTIMAL FILTER EQUATIONS

Our starting point in this section will be a statement of the suboptimal filter equations of Reference (3). A slightly simplified version is as follows:

Suppose we partition the state equations as:

$$\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} = \begin{bmatrix} \emptyset_{11} & \emptyset_{12} \\ \emptyset_{21} & \emptyset_{22} \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} + \begin{bmatrix} \underline{z}_1 \\ \underline{z}_2 \end{bmatrix} \tag{8}$$

and the measurement as:

$$\begin{bmatrix} \underline{y}_1 \\ \underline{y}_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix} + \begin{bmatrix} \underline{w}_1 \\ \underline{w}_2 \end{bmatrix} \tag{9}$$

The partitions $\underline{x}_1$ and $\underline{x}_2$ are then estimated in two separate filters. The equations for the estimate of $\underline{x}_1$ are:

$$\hat{\underline{x}}_1(t_{k+1}|k) = \sum_{i=1}^{2} \emptyset_{1i}(t_k)\ \hat{\underline{x}}(t_k|k) \tag{10-a}$$

$$\hat{\underline{x}}_1(t_{k+1}|k+1) = \hat{\underline{x}}_1(t_{k+1}|k) + A_1(t_{k+1}) \left[\underline{y}_1(t_{k+1}) - \sum_{i=1}^{2} M_{1i}\hat{\underline{x}}_i(t_{k+1}|k)\right] \tag{10-b}$$

where

$$A_1(t_{k+1}) = J_1(t_k)\ M_{11}^t\ (t_k) \left[M_{11}\ J_1\ M_{11}^t + W_1\right]^{-1} \tag{10-c}$$

and

$$J_1(t_k) = \emptyset_{11}(t_{k-1})\ J_1^+(t_{k-1})\ \emptyset_{11}^t(t_{k-1}) + Z_1(t_{k-1}) \tag{10-d}$$

$$J_1^+(t_k) = \left[I - A_1\ M_{11}\right] J_1(t_k) \tag{10-e}$$

A similar set of equations is used to estimate $\hat{\underline{x}}_2$. Of course, the procedure generalizes directly to more than two partitions of the state and measurement vector.

Observe that the equations (10) are defined on a purely formal basis, imitating the optimal equations. The extent to which the estimate so determined will approximate the optimal estimate will depend on the extent to which the filter equations can actually be uncoupled. For example, if M_{12}, M_{21}, ϕ_{12}, ϕ_{21}, W_{12} and Z_{12} are all zero and the apriori error covariance matrix is block diagonal, then the suboptimal estimate will actually be optimal.

An additional degree of freedom may be introduced in the above equations by choosing the matrices W_1 and W_2 to be something other than simply partitions of W. The manner usually used to do this is the following: suppose W is partitioned as:

$$W = \begin{bmatrix} W_{11} & W_{12} \\ W_{12}^t & W_{22} \end{bmatrix},$$

and we let

$$W_i = W_{ii} + k_i I, \; i = 1, 2.$$

The k_i are constants which we wish to choose so that the filter estimates are improved. This we do essentially empirically as follows:

- Choose values (arbitrarily) for k_1 and k_2
- Compute the actual error covariance matrix of the filter using Equations (5) and (7) with

$$W_i = W_{ii} + k_i I, \qquad i = 1, 2.$$

in the expressions for the filter gains.

- Repeat this for several values of k_1 and k_2, and choose those values which give the smallest deviation from the optimal error covariance matrix when all measurements have been processed.

This procedure is time consuming and aesthetically unsatisfying. We will now discuss a modification which will eliminate or significantly reduce this cut-and-try approach to "sub-optimally optimizing." The modification consists essentially in introducing coupling between the filters in the computation of the gains A_i based on computing orthogonal projections on subspaces of the linear space Y(k) defined in Part II.

Assume at some point in the filtering process we have an optimal estimate $\hat{\underline{x}}(t_k|k-1)$. We wish to process the measurement $\underline{y}(t_k)$ suboptimally, i.e. in two filters of dimension smaller than the optimal. We make the following assumption: the estimate $\hat{\underline{x}}(t_k|k-1)$ can be partitioned as:

$$\hat{\underline{x}}(t_k|k-1) = \begin{bmatrix} \hat{\underline{x}}_1 \\ \hat{\underline{x}}_2 \end{bmatrix} ,$$

so that

$$E\left\{\underline{\epsilon}_1(t_k|k-1)\ \underline{\epsilon}_2^t(t_k|k-1)\right\} \simeq 0.$$

i.e., the error covariance matrix $J(t_k)$ is assumed to have the form:

$$J(t_k) \simeq \begin{bmatrix} J_1(t_k) & 0 \\ 0 & J_2(t_k) \end{bmatrix} .$$

In addition, we partition the measurement $\underline{y}(t_k)$ as:

$$\underline{y}(t_k) = \begin{bmatrix} \underline{y}_1(t_k) \\ y_2(t_k) \end{bmatrix}$$

The manner in which this partition is arrived at will be discussed shortly.

Consider now the following four subspaces of $Y(k)$:

$$Y_1(k) = \left\{\underline{x}: \underline{x} \in Y(k),\ \underline{x} \text{ has the form: } \underline{x} = \sum_{i=0}^{k} \begin{bmatrix} B(i) & 0 \\ 0 & 0 \end{bmatrix} \underline{y}(t_i),\right.$$

$$\left. B(i) \text{ are arbitrary, real matrices of dimension } n_{x_1} \times n_{y_1}\right\}$$

$$Y_2(k) = \left\{\underline{x}: \underline{x} \in Y(k),\ \underline{x} \text{ has the form: } \underline{x} = \sum_{i=0}^{k} \begin{bmatrix} 0 & 0 \\ 0 & C(i) \end{bmatrix} \underline{y}(t_i) ,\right.$$

$$\left. C(i) \text{ an arbitrary, real matrices of dimension } n_{x_2} \times n_{y_2}\right\}.$$

$$Z_1(k) = \left\{\underline{x}: \underline{x} \in Y(k),\ \underline{x} \text{ has the form: } \underline{x} = \begin{bmatrix} A_1 & 0 \\ 0 & 0 \end{bmatrix} \left[\underline{y}(t_k)\right.\right.$$

$$\left.\left. - M(t_k)\ \hat{\underline{x}}(t_k|k-1)\right],\ A_1 \text{ real, arbitrary, of dimension } n_{x_1} \times n_{y_1}\right\}$$

$$Z_2(k) = \left\{\underline{x}: \underline{x} \in Y(k),\ \underline{x} \text{ has the form: } \underline{x} = \begin{bmatrix} 0 & 0 \\ 0 & A_2 \end{bmatrix} \left[\underline{y}(t_k)\right.\right.$$

$$\left.\left. - M(t_k)\ \hat{\underline{x}}(t_k|k-1)\right],\ A_2 \text{ real, arbitrary, of dimension } n_{x_2} \times n_{y_2}\right\}$$

Then, $Z_1(k)$ and $Z_2(k)$ are the orthogonal complements of $Y_1(k-1)$ and $Y_2(k-1)$ relative to $Y_1(k)$ and $Y_2(k)$, respectively.

We will now determine our suboptimal estimate $\begin{bmatrix} \hat{\underline{x}}_1(t_k|k) \\ \hat{\underline{x}}_2(t_k|k) \end{bmatrix}$ as:

$$\begin{bmatrix} \hat{\underline{x}}_1(t_k|k) \\ \hat{\underline{x}}_2(t_k|k) \end{bmatrix} = \begin{bmatrix} \hat{\underline{x}}_1(t_k|k-1) \\ \hat{\underline{x}}_2(t_k|k-1) \end{bmatrix} + \begin{bmatrix} A_1^* & 0 \\ 0 & A_2^* \end{bmatrix} \left[\underline{y}(t_k) - M\hat{\underline{x}}(t_k|k-1) \right] \tag{11}$$

where:

$\begin{bmatrix} A_1^* & 0 \\ 0 & 0 \end{bmatrix} \left[\underline{y}(t_k) - M\,\hat{\underline{x}}(t_k|k-1) \right]$ is the orthogonal projection of $\begin{bmatrix} \underline{x}_1(t_k) \\ 0 \end{bmatrix}$ on $Z_1(k)$, and $\begin{bmatrix} 0 & 0 \\ 0 & A_2^* \end{bmatrix} \left[\underline{y}(t_k) - M\,\hat{\underline{x}}\,(t_k|k-1) \right]$ is the orthogonal projection of $\begin{bmatrix} 0 \\ \underline{x}_2(t_k) \end{bmatrix}$ on $Z_2(k)$. Observe that we are neglecting the projections of $\begin{bmatrix} \underline{x}_1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 0 \\ \underline{x}_2 \end{bmatrix}$ on $Z_2(k)$ and $Z_1(k)$ respectively. The partition of $\underline{y}(t_k)$, or equivalently the spaces $Y_1(k)$ and $Y_2(k)$, should consequently be chosen so that these neglected projections are small.

The matrices A_1^* and A_2^* are calculated straightforwardly as follows: Let $\underline{y}(t_k) - M\,\hat{\underline{x}}(t_k|k-1) = \underline{y}'(t_k) = \text{col}\left\{y_1', \ldots y_{n_{y_1}}', \ldots y_{n_y}'\right\}$ (where "col $\{\ \}$ denotes a column array). Then since $\begin{bmatrix} A^* & 0 \\ 0 & 0 \end{bmatrix} \underline{y}'(t_k)$ is the orthogonal projection of $\begin{bmatrix} \underline{x}_1 \\ 0 \end{bmatrix}$ on $Z_1(k)$, the quantity $\begin{bmatrix} \underline{x}_1 \\ 0 \end{bmatrix} - \begin{bmatrix} A_1^* & 0 \\ 0 & 0 \end{bmatrix} \underline{y}'(t_k)$ is orthogonal to every vector in $Z_1(k)$, and in particular, to a basis for $Z_1(k)$. But the vectors $\text{col}\left\{y_1', 0, \ldots 0\right\}$, $\text{col}\left\{y_2', 0, \ldots 0\right\}$, ..., $\text{col}\left\{y_{n_{y_1}}', 0, \ldots 0\right\}$, $\text{col}\left\{0, y_1', 0, \ldots 0\right\}$, ..., $\text{col}\left\{0, \ldots 0, y_{n_{y_1}}', 0, \ldots 0\right\}$, constitute a basis for $Z_1(k)$. Hence, the matrix A_1^* satisfies the relation:

$$E\left[\left\{\underline{x}_1 - \left[A_1^*\ 0\right]\left[\underline{y} - M\,\hat{\underline{x}}\right]\right\}\left\{\underline{y}_1 - \sum_{i=1}^{2} M_{1i}\hat{\underline{x}}(t_k|k-1)\right\}^t\right] = 0 \tag{12}$$

Writing $\underline{x}_1(t_k) = \hat{\underline{x}}_1(t_k|k-1) + \underline{\epsilon}_1(t_k|k-1)$, and recalling that the estimate $\hat{\underline{x}}(t_k|k-1)$ is a member of $Y(k-1)$ and hence orthogonal to $Z_1(k)$ and $Z_2(k)$, we get from (12):

$$A_1^* = E\left\{\underline{\epsilon}_1\left[\underline{y}_1 - \sum_{i=1}^{2} M_{1i}\hat{\underline{x}}_i\right]^t\right\}\left[E\left[\underline{y}_1 - \sum_{i=1}^{2} M_{1i}\hat{\underline{x}}_i\right]\left[\underline{y}_1 - \sum_{i=1}^{2} M_{1i}\hat{\underline{x}}_i\right]^t\right]^{-1}$$

For which we get, recalling that, by assumption $E\left\{\underline{\epsilon}_1\ \underline{\epsilon}_2^t\right\} = 0$,

$$A_1^*(t_k) = J_1(t_k)\, M_{11}^t(t_k) \Big[M_{11}(t_k) J_1(t_k)\, M_{11}^t(t_k) + W_{11}(t_k) + M_{12}(t_k) J_2(t_k)\, M_{12}^t(t_k) \Big]^{-1} \qquad (13)$$

A similar expression is obtained for A_2^*. Observe that A_1^* and A_2^* will each contain a term which depends on the error covariance matrix of the estimate in the other filter.

The suboptimal filter equations proposed here will thus have the same form as those of Equations 10-a through 10-e, with the exception of the gain Equation 10-c. This will be given by equations of the form (13).

It should be pointed out that no procedure other than intuitive reasoning is given here for the manner in which partitions of the state and measurement vectors are to be chosen; this may frequently represent a severe limitation on the technique proposed, since it may happen that such partitions are not intuitively obvious.

We proceed now to a discussion of an application of this procedure to a practical problem.

IV. EXAMPLE

In this section we will compare the performance of an optimal filter with two suboptimal schemes; the first suboptimal filter uses Equations 10-a - e, with matrices W_1 and W_2 selected empirically as described in the previous section. The second filter calculates the gains from Equation (13).

The geometry of the problem we are considering is shown in Figure I. An airplane flies north for a distance of 30 nautical miles at a speed of 600 knots. The pilot takes bearing measurements at ten equally spaced time intervals on a stationary radiating target. The pilot has a navigation system to determine his own position, and has initial position and velocity uncertainties. His angle measurements have a constant bias, and are contaminated with additive white noise. He will process the bearing measurements in an optimal (linearized) filter and estimate the target coordinates.

The observations are thus the airplane coordinates x_A^* and y_A^* generated by the airplane navigation system, and the target bearing Θ^*.

An aircraft navigational system is assumed to relate output coordinates x_A^* and y_A^* to actual coordinates x_A and y_A by:

$$\begin{aligned} x_A^* &= x_A + B_x + w_x \\ y_A^* &= y_A + B_y + w_y \end{aligned} \qquad (14)$$

B_x and B_y are (constant) biases. w_x and w_y represent the (white) measurement noise.

The measured bearing Θ^* is related to the target and airplane coordinates as:

$$\theta^* = \tan^{-1}\left\{\frac{y_T - y_A}{x_T - x_A}\right\} + \theta_B + w_\theta \tag{15}$$

θ_B is a constant bias; w_θ is the(white) measurement noise on the angle measurement.

We will use a linearized Kalman filter to estimate the coordinates x_A, y_A, x_T, y_T, velocities $\dot{x}_A$, $\dot{y}_A$, and biases B_x, B_y and θ_B, using measurements (14) and (15).

The state equations are (assuming no state noise):

$$\begin{bmatrix} x_A(t_{k+1}) \\ \dot{x}_A(t_{k+1}) \\ y_A(t_{k+1}) \\ \dot{y}_A(t_{k+1}) \\ x_T(t_{k+1}) \\ y_T(t_{k+1}) \\ B_x(t_{k+1}) \\ B_y(t_{k+1}) \\ \theta_B(t_{k+1}) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & & & & & & & \\ 0 & 1 & & & & & & & \\ & & 1 & \Delta t & & & & & \\ & & 0 & 1 & & & & & \\ & & & & 1 & 0 & & & \\ & & & & 0 & 1 & & & \\ & & & & & & 1 & 0 & 0 \\ & & & & & & 0 & 1 & 0 \\ & & & & & & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_A(t_k) \\ \dot{x}_A(t_k) \\ y_A(t_k) \\ \dot{y}_A(t_k) \\ x_T(t_k) \\ y_T(t_k) \\ B_x(t_k) \\ B_y(t_k) \\ \theta_B(t_k) \end{bmatrix}$$

We observe that Equation (15) is non-linear in the target and airplane coordinates. We will consequently linearize this equation about the nominal straight line trajectory, and estimate perturbations about the straight line.

The state vector was partitioned as:

$$\underline{x}_1 = \text{col}\left\{x_A, \dot{x}_A, x_T, y_T, B_x\right\}, \ \underline{x}_2 = \text{col}\left\{y_A, \dot{y}_A, B_y, \theta_B\right\}$$

And the measurement vector as:

$$\underline{y}_1 = \text{col}\left\{x_A^*, \theta^*\right\}, \ \underline{y}_2 = \text{col}\left\{y_A^*, \theta^*\right\}$$

The filter gains for the optimal and suboptimal filter were calculated, and used in Equation (5) and (7) to calculate the error covariance matrices. The variance in the error in the estimate of the target x and y coordinates at the termination of the maneuver is shown in Figures II-(a) and (b). One sigma error ellipses for the "best" suboptimal filter without coupling, the suboptimal filter with coupling, and the optimal filter are shown in Fig. III. Figures II and III indicate that the addition of the coupling term improves the performance of the "empirically optimized" suboptimal filter, at the expense of the slightly greater computational requirement to determine the coupling term.

V. CONCLUDING REMARKS

A method has been presented for the design of suboptimal sequential estimators which uses a formally similar set of equations as Reference (3), but calculates the filter weights by computing orthogonal projections on subspaces of the observation space used in the optimal filter. The motivation is to effect a simplification of the optimum filter; the suboptimal filters which are generated by this technique achieve this end in that they involve the manipulation of matrices of dimension smaller than those which appear in the optimal filter.

The design procedure yields an expression for the suboptimal filter gains which has a form similar to that for the optimal filter, but contains a term (or terms)arising from the uncoupling procedure which effectively increases the covariance of the observation noise when the filter weights are calculated.

The central assumption in the suboptimization technique is that the state vector can be partitioned into subvectors which are (essentially) pairwise uncorrelated. If the system dynamics are such that this assumption is not satisfied, the filter designer should look for a coordinate change so that, to a better approximation, the state vector may be so partitioned. Engineering intuition is probably the most useful way to obtain such a coordinate change.

An example was presented which compares the performance of a filter designed by this technique with the optimal filter, and with another suboptimal filter for which the gains are determined by a cut-and-try technique; the performance of the first suboptimal filter was superior to the second.

Acknowledgements

The author would like to thank Mr. Paul Chen and Mr. Albert Monuki for reading this manuscript, and Mrs. Norma Cooley for typing it.

References

(1) Kalman, R.E., "A New Approach to the Linear Filtering and Prediction Problem," Journal of Basic Engineering, March 1960.

(2) Joseph, P.D., "On-Board Navigation for Rendezvous Missions."

(3) Joseph, P.D., "Suboptimal Linear Filtering," STL Interoffice Correspondence No. 9321.4-653.

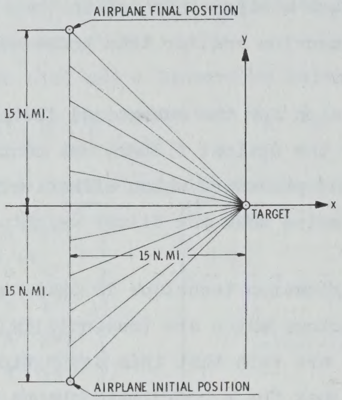

AIRPLANE FINAL POSITION
y
15 N. MI.
x
TARGET
15 N. MI.
15 N. MI.
AIRPLANE INITIAL POSITION

FIGURE 1 PROBLEM GEOMETRY

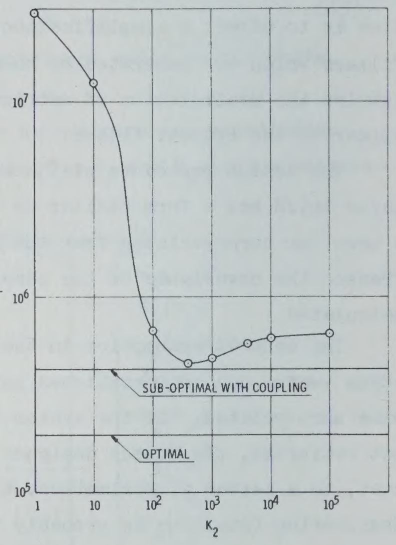


FIGURE 2A x_T VARIANCE VS K_2

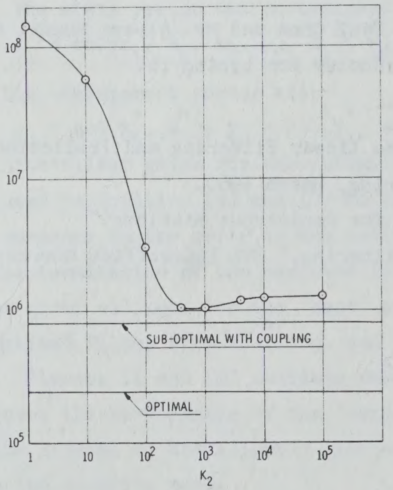


FIGURE 2B y_T VARIANCE VS K_2

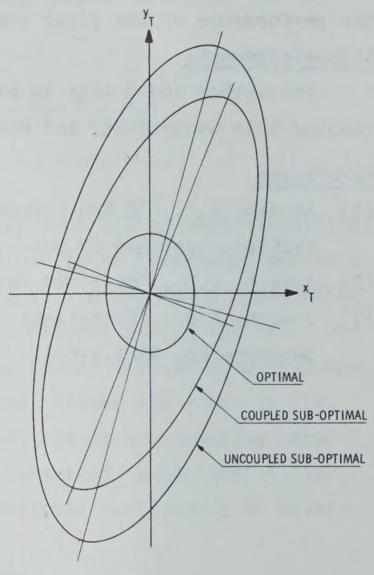

y_T
x_T
OPTIMAL
COUPLED SUB-OPTIMAL
UNCOUPLED SUB-OPTIMAL

FIGURE 3 ERROR ELLIPSES FOR OPTIMAL, COUPLED SUB-OPTIMAL AND UNCOUPLED SUB-OPTIMAL FILTERS

STUDIES IN DYNAMIC FILTERING

WILLIE L. MCDANIEL, JR.
Electrical Engineering Department
Mississippi State University, State College, Mississippi

ABSTRACT

The theories developed in recent years concerning optimal filtering, estimation, and prediction have been used in varying degrees for determining trajectory solutions and for designing navigational systems. This paper treats two purely academic problems in an effort towards a better understanding of the use of "digital" filters in their real world applications.

The two problems treated are "post-analysis" problems rather than "real-time" problems. A basic Kalman filtering approach is taken; however, the problems are simple enough to give a clear understanding of the step by step recursive nature of the technique. The model in each problem is that of a linear system with additive random errors in the observations.

Problem One

This problem involves a block moving with constant velocity on a frictionless surface with "noisy" observations being made on its position. The necessary statistics concerning the noise and an initial position estimate and covariance matrix of the error in this estimate are known. The problem is to estimate the position and velocity of the block. This simple problem can be solved by hand and gives insight into the Kalman filtering equations.

Problem Two

Consider the case of a unit mass moving with a constant areal velocity in a circular path of unit radius. Observations corrupted by noise are made on position. An initial estimate, the covariance matrix of the error in this estimate, and the statistics of the noise are known. The problem is to estimate a four-element state vector composed of position and velocity components.

CONCLUSION

The two problems considered are simple physically but complicated enough to give an appreciation for linear optimal estimation theory. It is hoped that some insight has been gained into the accuracy of the selection of the initial state estimate, the covariance matrix of the error in this estimate and its relationship to the covariance matrix of the noise.

STOCHASTIC APPROXIMATION NONPARAMETRIC TRAINING PROCEDURES FOR MULTI-CATEGORY PATTERN CLASSIFIERS

J. M. SCHUMPERT
Science Services Division
Texas Instruments, Inc.
Dallas, Texas

S. S. YAU
Department of Electrical Engineering
Northwestern University
Evanston, Illinois

SUMMARY

In this paper, two nonparametric training procedures for multi-category pattern classifiers are proposed. These two training procedures are based on the Robbins-Monro stochastic approximation procedure. The computation required is very simple. The iterative property of these two procedures enables the pattern classifier to be adaptive when the pattern classes are time-variant. Several examples are simulated and their results converge to local optima.

I. INTRODUCTION

In the design of pattern classifiers, one can easily use statistical decision theory to find an optimal pattern classifier (in the Bayes sense) when the form of the probability distribution of the patterns in each category is known. However, the information about the pattern classes available to the designer is usually only a set of sample patterns which is assumed to be a good representation of the pattern classes. In this case, it is often possible to achieve a reasonable degree of success by using discriminant functions of fixed form with a set of adjustable weights. The procedure for obtaining a suitable set of weights for such discriminant functions directly from the sample patterns is called nonparametric training procedure [1].

Assume there are R pattern classes, and let a pattern be represented by a q-dimensional column vector $\underline{x}=(x_1,x_2,\ldots x_q)'$, where the prime of a matrix denotes its transpose, and the components of $\underline{x}$ correspond to the measurements of the pattern. The discriminant functions $\{g_i(\underline{x})|1\leq i\leq R\}$ considered in this paper have the form

$$g_i(\underline{x}) = \sum_{k=1}^{p} w_{ik}\phi_k(\underline{x}), \qquad (1)$$

where the real-valued functions $\{\phi_k|1\leq k\leq p\}$ are chosen according to some reasonable criterion and $\{w_{ik}|1\leq i\leq R, 1\leq k\leq p\}$ are a set of adjustable weights [1]. For example, the ϕ_k's may be polynomials of the components of $\underline{x}$. When the weights $\{w_{ik}|1\leq i\leq R, 1\leq k\leq p\}$ are properly adjusted, we can use the following decision rule for classification:

$$\underline{x} \text{ is classified in Category } \theta^i \text{ if } g_i(\underline{x}) = \max_{1\leq j\leq R} g_j(\underline{x}). \qquad (2)$$

The effectiveness of this classification scheme, of course, depends upon the values of the w_{ik}'s as well as the ϕ_k's. We shall assume that the ϕ_k's are specified and present two training algorithms for finding the w_{ik}'s, which are based on a stochastic approximation technique.

Recently, several authors [2-5] have used stochastic approximation and similar algorithms as training procedures for multi-category pattern classification. These results depended on the assumption that the a posteriori probability $P(\theta^i|\underline{x})$ of Category θ^i, given the pattern $\underline{x}$, could be approximated by the linear combination

$$P(\theta^i|\,\underline{x}) \approx \sum_{k=1}^{m} a_{ik}h_k(\underline{x}), \quad i=1,2,\ldots,R, \qquad (3)$$

where $\{h_k|1\leq k\leq m\}$ are some predetermined functions of $\underline{x}$. Under this assumption, the a_{ik}'s were found by a stochastic approximation algorithm. Unfor-

---

This work was supported by the Joint Services Electronics Program (U. S. Army, U. S. Navy and U. S. Air Force) under Office of Naval Research Contract Number N00014-67-A-0356-0003 Modification AB.

tunately, there is no well-defined procedure for selection of the h_k's. This is in contrast to the criteria for selecting the ϕ_k's of (1) since these are usually chosen as polynomials in $\underline{x}$ [1,5]. Also it is quite likely that the functions $\{h_k\}$ will be too complicated to realize in a physical system.

We shall propose two training procedures which avoid the above difficulties. These procedures are used for processing an arbitrary sequence of sample patterns and result in a set of weights for the discriminant functions $\{g_i(\underline{x}) | 1 \leq i \leq R\}$ in (1). Because of this property, the training procedures can update the weights whenever new sample patterns are available. The algorithms are based on the Robbins-Monro stochastic approximation procedure [6] which will briefly be described in the next section.

II. THE ROBBINS-MONRO PROCEDURE

The Robbins-Monro procedure is a scheme for locating the root of an unknown regression function. We shall illustrate how this method can be used as a training procedure for pattern classification problems.

Let us write the discriminant functions of (1) as

$$g_i(\underline{x}) = \underline{w}_i' \, \underline{\phi}(\underline{x}), \qquad i = 1,\ldots,R, \tag{4}$$

where

$$\underline{w}_i' = [w_{i1},\ldots,w_{ip}], \qquad i = 1,\ldots,R, \tag{5}$$

$$\underline{\phi}'(\underline{x}) = [\phi_1(\underline{x}),\ldots,\phi_p(\underline{x})], \tag{6}$$

and form the Rp-dimensional weight vector

$$\underline{w} = \begin{bmatrix} \underline{w}_1 \\ \vdots \\ \underline{w}_R \end{bmatrix}. \tag{7}$$

Let $f(\underline{w},\underline{x})$ be a real-valued function of $\underline{w}$ and $\underline{x}$. Let $G(\underline{w})$ be the expectation of $f(\underline{w},\underline{x})$ with respect to $\underline{x}$, i.e., if $p(\underline{x})$ is the probability density function of $\underline{x}$,

$$G(\underline{w}) = E\{f(\underline{w},\underline{x})\} = \int_{E^q} f(\underline{w},\underline{x})\, p(\underline{x})\, d\underline{x}, \tag{8}$$

which is a real-valued function of $\underline{w}$ and whose maximum or minimum with respect to $\underline{w}$ is used as a criterion for optimization. We assume that $p(\underline{x})$ is unknown, and $G(\underline{w})$ is to be maximized with respect to $\underline{w}$. Then, if $G(\underline{w})$ has a unique maximum at $\underline{w}=\hat{\underline{w}}$, and if certain other conditions [6,7] are satisfied, the following iterative procedure will produce a sequence $\{\underline{w}_n\}$ which converges to $\hat{\underline{w}}$ with probability one:

Let $\{\underline{x}_n\}$ be a sequence of independent and identically distributed random variables and $\{a_n\}$ a sequence of positive real numbers which satisfy the following conditions:

$$\sum_{n=1}^{\infty} a_n = \infty, \qquad \sum_{n=1}^{\infty} a_n^2 < \infty. \tag{9}$$

Choose $\underline{w}_1$ arbitrarily and define recursively

$$\underline{w}_{n+1} = \underline{w}_n + a_n \nabla f(\underline{w}_n,\underline{x}_n), \tag{10}$$

where ∇ denotes the gradient operator.

The algorithm represented by (10) is a Robbins-Monro procedure in locating a root of $\nabla G(\underline{w})=\underline{0}$. In the following section we shall show how this algorithm may be used as a training procedure for pattern classification.

III. THE TRAINING PROCEDURES

Consider the discriminant functions $\{g_i(\underline{x}) | 1 \leq i \leq R\}$ of the form given in (4). If $p=q+1$, and ϕ_k's are defined as

$$\begin{aligned} \phi_k(\underline{x}) &= x_k, \qquad k = 1,2,\ldots,q, \\ \phi_p(\underline{x}) &= 1, \end{aligned} \tag{11}$$

then the $g_i(\underline{x})$, $i=1,2,\ldots,R$, are the well-known linear discriminant functions. Since the $g_i(\underline{x})$ given in (4) always linearly depends on the weights w_{ik}, $k=1,2,\ldots p$, regardless of the forms of $\phi_k(\underline{x})$, the training procedures applicable to linear discriminant functions are also applicable to discriminant functions of other types of $\phi_k(\underline{x})$ [1]. Hence, we shall only discuss linear discriminant functions.

The decision rule will be that given in (2). Thus, it is desirable to find a set of weight vectors $\underline{w}_1,\ldots,\underline{w}_R$ such that for each i, $g_i(\underline{x})$ is "large" over the region where the density $p_i(\underline{x})$ is "large" while each $g_j(\underline{x})$, $j\neq i$, is "small" over the same region. Consider, therefore, the function

$$f_i(\underline{x},\underline{w}) = \frac{\exp(g_i(\underline{x}))}{\sum_{j=1}^{R} \exp(g_j(\underline{x}))} \qquad i=1,\ldots,R, \tag{12}$$

where $\underline{w}$ is given by (7). Note that $0\leq f_i(\underline{x},\underline{w})\leq 1$ for all $\underline{x}$ and $\underline{w}$ and $i=1,\ldots,R$. Also when $f_i(\underline{x},\underline{w})$ is close to 1 over the region where $p_i(\underline{x})$ is large, then $g_i(\underline{x})$ will be large with respect to $\max_{j\neq i} g_j(\underline{x})$. Another property of the f_i's is that the decision rule (2) is equivalent to:

$$\underline{x} \text{ is classified in Category } \theta^i \text{ if } f_i(\underline{x},\underline{w}) = \max_{1\leq j\leq R} f_j(\underline{x}). \tag{13}$$

Hence, during the training process, we can adjust the weight vectors according to the relative magnitude of the f_i's and then use the decision rule (2) after the training process has terminated.

A. The First Training Procedure

The first training procedure is based on the maximization of

$$G_1(\underline{w}) = G_1(\underline{w}_1,\ldots,\underline{w}_R) = \sum_{i=1}^{R} q_i E_i\{f_i(\underline{x},\underline{w})\}, \tag{14}$$

where q_i and E_i are the a priori probability of Category θ^i and the expectation operator with respect to $p_i(\underline{x})$, respectively.

It can be seen that $G_1(\underline{w})$ has no unique maximum. Thus, a Robbins-Monro procedure for maximizing $G_1(\underline{w})$ will yield only a local maximum of $G_1(\underline{w})$. However, we can start the algorithm at the vector $\tilde{\underline{w}}$ corresponding to a minimum-distance classifier [1] and obtain a classifier whose performance appears to be between that of an optimal and a minimum-distance classifier.

Let $\Theta=\{\theta^1,\ldots,\theta^R\}$ denote the set of categories and define the random variables $d_i(\theta)$ $(1\leq i\leq R)$ on Θ as follows:

$$d_i(\theta) = \begin{cases} 1 & \text{if } \theta = \theta^i \\ 0 & \text{otherwise}. \end{cases} \tag{15}$$

Let

$$f(\underline{x},\theta,\underline{w}) = \sum_{i=1}^{R} d_i(\theta) f_i(\underline{x},\underline{w}), \tag{16}$$

where the f_i's are given in (12). Then, it can be shown that

$$E\,\{f(\underline{x},\theta,\underline{w})\} = \sum_{i=1}^{R} q_i E_i\{f_i(\underline{x},\underline{w})\}$$

$$\triangleq G_1(\underline{w}) \ . \tag{17}$$

We shall use the Robbins-Monro procedure to find a solution of

$$\nabla G_1(\underline{w}) = \underline{0} \ . \tag{18}$$

Let

$$\underline{v}_1(\underline{x},\theta,\underline{w}) = \nabla f(\underline{x},\theta,\underline{w}). \tag{19}$$

Computing $\nabla f_i(\underline{x},\underline{w})$ and suppressing the arguments of the f_i's, we obtain the $R(q+1)$-dimensional vector

$$\nabla f_i(\underline{x},\underline{w}) = \begin{bmatrix} -f_i f_1 \ \underline{\phi}(\underline{x}) \\ \vdots \\ f_i(1-f_i) \ \underline{\phi}(\underline{x}) \\ \vdots \\ -f_i f_R \ \underline{\phi}(\underline{x}) \end{bmatrix} . \tag{20}$$

Hence, by (17) we have

$$\underline{v}_1(\underline{x},\theta,\underline{w}) = \begin{bmatrix} f_1[d_1(\theta)(1-f_1) - \sum\limits_{\substack{j=1 \\ j\neq 1}}^{R} d_j(\theta)f_j] \ \underline{\phi}(\underline{x}) \\ \vdots \\ f_R[d_R(\theta)(1-f_R) - \sum\limits_{\substack{j=1 \\ j\neq R}}^{R} d_j(\theta)f_j] \ \underline{\phi}(\underline{x}) \end{bmatrix} \tag{21}$$

and the Robbins-Monro procedure for solution of (18) is

$$\underline{w}(n+1) = \underline{w}(n) + a_n \ \underline{v}_1(\underline{x}_n,\theta_n,\underline{w}(n)) \ , \tag{22}$$

where $\underline{x}_1,\ldots,\underline{x}_n,\ldots$ is a sequence of sample patterns with $\theta_n = \theta^i$ if $\underline{x}_n$ belongs to Category θ^i. To obtain a recursive relation for each weight vector $\underline{w}_i$ from (22), define

$$\Delta_i(n) = f_i[d_i(\theta_n)(1-f_i) - \sum_{\substack{j=1 \\ j\neq i}}^{R} d_j(\theta_n)f_j]. \tag{23}$$

Then, for $i=1,\ldots,R$ we have

$$\underline{w}_i(n+1) = \underline{w}_i(n) + a_n \Delta_i(n) \ \phi(\underline{x}_n) \ . \tag{24}$$

In order to use (24) as a training procedure, we set

$$\underline{w}_i(1) = \begin{bmatrix} \tilde{\underline{w}}_i \\ -\frac{1}{2} \| \tilde{\underline{w}}_i \|^2 \end{bmatrix} \qquad i=1,\ldots,R, \tag{25}$$

where the vectors $\tilde{\underline{w}}_i$ are the q-dimensional weight vectors corresponding to a minimum-distance classifier.

B. The Second Training Procedure

The second training procedure deals with the function

$$E \ \{ \sum_{i=1}^{R} [P(\theta^i|\underline{x}) - f_i(\underline{x},\underline{w})]^2 \} \tag{26}$$

which we wish to minimize with respect to $\underline{w}$. Minimizing (26) is, in a sense, a criterion for the quality of the approximations of $P(\theta^i|\underline{x})$ by $f_i(\underline{x},\underline{w})$, $1 \le i \le R$. We cannot minimize (26) directly since $P(\theta^i|\underline{x})$ cannot be observed; however, we can use stochastic approximation for minimizing

$$G_2(\underline{w}) = E\{\sum_{i=1}^{R} [d_i(\theta) - f_i(\underline{x},\underline{w})]^2\}. \tag{27}$$

It can easily be shown that minimizing $G_2(\underline{w})$ is equivalent to minimizing (26).

We shall use a Robbins-Monro procedure for locating a root of

$$\nabla G_2(\underline{w}) = \underline{0}. \tag{28}$$

Let

$$\underline{v}_2(\underline{x},\theta,\underline{w}) = \frac{1}{2} \nabla \{\sum_{i=1}^{R} [d_i(\theta) - f_i(\underline{x},\underline{w})]^2\}. \tag{29}$$

Then using the notation of (20) and suppressing the arguments of the f_i's, we have

$$\underline{v}_2(\underline{x},\theta,\underline{w}) = \begin{bmatrix} f_1[(d_1(\theta) - f_1)(1-f_1) - \sum_{\substack{j=1 \\ j\neq 1}}^{R} (d_j(\theta) - f_j)f_j]\, \underline{\phi}(\underline{x}) \\ \vdots \\ f_R[(d_R(\theta) - f_R)(1-f_R) - \sum_{\substack{j=1 \\ j\neq R}}^{R} (d_j(\theta) - f_j)f_j]\, \underline{\phi}(\underline{x}) \end{bmatrix}. \tag{30}$$

Thus, the Robbins-Monro procedure for location of a local maximum of $G_2(\underline{w})$ is given by

$$\underline{w}(n+1) = \underline{w}(n) - a_n \underline{v}_2(\underline{x}_n,\theta_n,\underline{w}(n)) , \tag{31}$$

where a_n, x_n, θ_n are as defined earlier. If we define for each i, $(1 \le i \le R)$,

$$\gamma_i(n) = - f_i[(d_i(\theta_n) - f_i)(1-f_i) - \sum_{\substack{j=1 \\ j\neq i}}^{R} (d_j(\theta_n) - f_j)f_j], \tag{32}$$

then the components of (31) may be separated yielding the following recursive relations for each weight vector:

$$\underline{w}_i(n+1) = \underline{w}_i(n) + a_n \gamma_i(n)\, \underline{\phi}(\underline{x}_n) \qquad i=1,\dots,R. \tag{33}$$

The training procedure is specified by (33) and the initial weight vectors $\underline{w}_i(1)$ given in (25).

In both of these methods we cannot be assured of convergence to the absolute extrema of G_1 and G_2 since these functions are not unimodal. However, it is reasonable to assume that $G_1(\underline{w})$ and $G_2(\underline{w})$ have local extremas which are respectively greater than $G_1(\underline{w}(1))$ and less than $G_2(\underline{w}(1))$ so that the algorithms converge to the weight vectors which perform better than those corresponding to minimum-distance classifiers.

C. Examples

The training procedures described above were tested on several 3-category, 2-dimensional examples, and the resulting classifiers were compared with the optimal Bayes classifiers (whose discriminant functions were computed from the actual distributions). The patterns were drawn from three Gaussian distributions having the same covariance matrix $\underline{\Sigma}$ but different means $\underline{\mu}_1$, $\underline{\mu}_2$, $\underline{\mu}_3$. 3000 patterns where randomly selected with a priori probabilities $q_1 = q_2 = 0.4$, $q_3 = 0.2$, and these formed the training sequence for generation of the weight vectors. The resulting classifiers were then compared to the Bayes classifiers by using 500 test patterns.

The results of these tests are summarized in Table I, and the decision surfaces are plotted in Figs. 1 and 2. Fig. 1 shows the decision surfaces for Examples 1 and 2 while Fig. 2 shows the decision surfaces for examples 3 and 4. The optimal (Bayes) decision surfaces are plotted in both figures.

IV. CONCLUSIONS

We have shown how stochastic approximation methods can be used as training procedures for multi-category pattern classifiers. The training procedures derived here are iterative and therefore allow updating of the classifiers' performance at each stage of the training sequence. Also these procedures are nonparametric, and the knowledge of the form of the distributions is not required. Application of these procedures results in classfiers whose performance is between that of minimum-distance and optimal classifiers.

REFERENCES

[1] N. J. Nilsson, *Learning Machines* , McGraw-Hill, New York, 1965.

[2] C. Blaydon and Y. C. Ho, "On the Abstraction Problem in Pattern Recognition", *Proc. 1966 National Electronics Conference*, vol. 22, Chicago, Ill., pp. 857-862, October,1966.

[3] Z. J. Nikolic and K. S. Fu, "A Mathematical Model of Learning in a Random Environment", *Proc. 1966 National Electronics Conference*, vol.22, Chicago, Ill., pp. 607-612, October, 1966.

[4] R. L. Kashyap and C. C. Blaydon, "Recovery of Functions from Noisy Measurements taken at Randomly Selected Points and its Application to Pattern Classification", *Proc. IEEE*, vol. 54, pp. 1127-1129, August 1966.

[5] G. S. Sebestyen, *Decision Making Processes in Pattern Recognition*, Macmillan, New York, 1962.

[6] H. Robbins and S.Monro, "A Stochastic Approximation Method", *Annals of Mathematical Statistics*, vol. 22, pp. 400-407, 1951.

[7] D. J. Sakrison, "Stochastic Approximation: A Recursive Method for Solving Regression Problems", in A. V. Balakrishman (ed.) *Advances in Communication Systems*, vol. 2, Academic Press, New York, pp. 51-106, 1966.

Table I.
Results of Tests Using Algorithms (24) and (33)

	Pattern Statistics				Classifier error-rate	Bayes error-rate
	$\underline{\Sigma}$	$\underline{\mu}_1$	$\underline{\mu}_2$	$\underline{\mu}_3$		
Example 1 Algorithm (24)	$\begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 3 \\ -3 \end{bmatrix}$	0.096	0.048
Example 2 Algorithm (33)	$\begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 3 \\ -3 \end{bmatrix}$	0.084	0.066
Example 3 Algorithm (24)	$\begin{bmatrix} 1 & 0 \\ 0 & 16 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 3 \\ -3 \end{bmatrix}$	0.198	0.106
Example 4 Algorithm (33)	$\begin{bmatrix} 1 & 0 \\ 0 & 16 \end{bmatrix}$	$\begin{bmatrix} 4 \\ 4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 3 \\ -3 \end{bmatrix}$	0.226	0.110

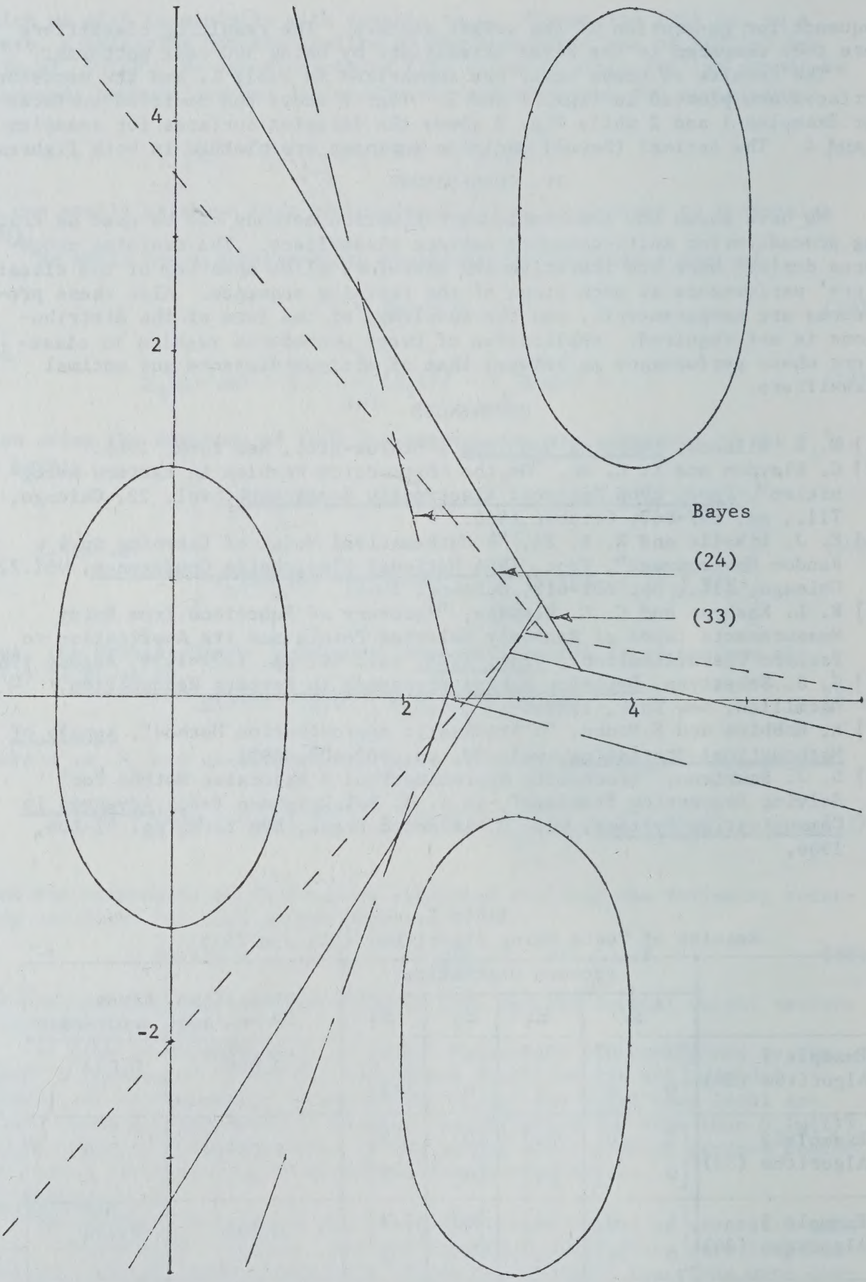

4
2
-2
2
4
Bayes
(24)
(33)

Figure 1
Decision Surfaces for Examples 1 and 2

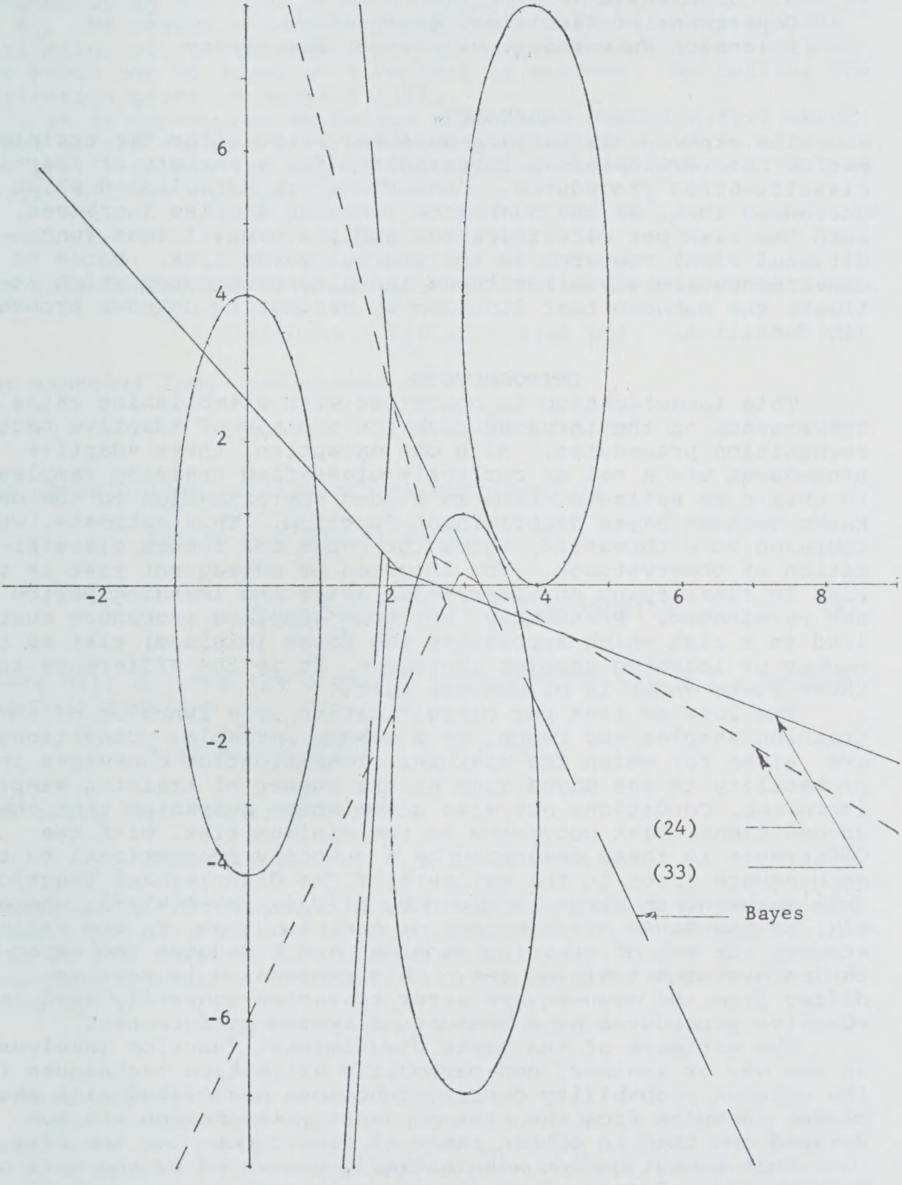


Figure 2

Decision Surfaces for Examples 3 and 4

CONVERGENCE OF RISK IN ADAPTIVE PATTERN RECOGNITION PROCEDURES

S. C. SCHWARTZ
Department of Electrical Engineering
Princeton University, Princeton, New Jersey

ABSTRACT

The risk in classifying an observation after the training period has terminated is investigated for a variety of adaptive classification procedures. Conditions are established which guarantee that, as the number of learning samples increases, both the risk per classification and the overall loss (unconditional risk) converge to the minimum Bayes risk. Rates of convergence are given for those learning procedures which estimate the unknown test function by estimating unknown probability densities.

INTRODUCTION

This investigation is concerned with establishing rates of convergence on the incurred risk for a class of adaptive pattern recognition procedures. With one exception, these adaptive procedures use a set of correctly classified training samples to obtain an estimate which is a good approximation to the unknown optimum Bayes discriminant function. This estimate, when compared to a threshold, forms the basis for future classification of observations. The incurred or subsequent risk is the risk in classifying an observation after the learning period has terminated. Presumably, any good adaptive procedure must lead to a risk which approaches the Bayes (minimum) risk as the number of learning samples increases. It is the difference in these risks which is of concern here.

The loss or risk per classification is a function of the training samples and hence, is a random variable. Conditions are given for which the risk per classification converges in probability to the Bayes risk as the number of training samples increases. Conditions are also given which guarantee that the unconditional risk converges to the minimum risk, with the difference in these dominated by a quantity proportional to the mean-square error in the estimate of the discriminant function. This mean-square error is given by $E\{(T_n(\chi_n;x)-T(x))^2\}$, where $T(x)$ is the Bayes discriminant or test function, T_n the estimate, χ_n the set of training samples, and E denotes the expectation over the training set. This expression is seen to differ from the mean-square error criterion generally used in adaptive procedures as a measure of system performance.

The estimate of the Bayes discriminant function involves, in one way or another, non-parametric estimation techniques for the unknown probability density functions associated with each class. Results from the area of density estimation are summarized and used to obtain rates of convergence for the risk.

This investigation was initially motivated by the work of Robbins [8]. Points of similarity with some recent work of Van Ryzin [11] are discussed.

STATEMENT OF THE PROBLEM

The case of scaler observations, known to belong to one of two classes π_0 or π_1, is considered. The density function of the observation is given by $f_\lambda(x)$, where $\lambda = 0,1$, designates the class. Let the a priori probabilities be p_0 and $p_1 = 1-p_0$. The loss incurred in misclassifying an observation from π_0 is denoted by a_0 and that of misclassifying an observation from π_1 by a_1. We assume zero loss for correct classifications. The derivation of the optimum Bayes procedure when all statistics are known can be found in a variety of sources. We outline the derivation given in Robbins ([8]).

It is convenient to define a decision function $t(x)$ which takes on the value 0 or 1, depending on which class we announce. Let $L(i,j)$ denote the loss in announcing class i when j is true (e.g., $L(1,0)= a_0$) and set

$$b(\lambda) = L(0,\lambda) - L(1,\lambda) \quad . \tag{1}$$

The loss can now be written in terms of the decision function as

$$L(t(x),\lambda) = L(0,\lambda) - t(x)\, b(\lambda) \quad . \tag{2}$$

The expected loss conditioned on λ is

$$R(t,\lambda) = \int_x L(t(x),\lambda)\, f_\lambda(x)\, dx \quad . \tag{3}$$

The overall loss or risk can be expressed as

$$\begin{aligned} R(t,\pi) &= \int_\lambda R(t,\lambda)\, d\pi(\lambda) \\ &= \int_\lambda \int_x L(t(x),\lambda)\, f_\lambda(x)\, dx d\pi(\lambda) \quad , \end{aligned} \tag{4}$$

where $\pi(\lambda)$ denotes the a priori distribution $\{p_0,p_1\}$. Defining the test function

$$T(x) = p_1 a_1 f_1(x) - p_0 a_0 f_0(x) \tag{5}$$

and using (2), the risk becomes

$$R(t,\pi) = p_1 a_1 - \int_x t(x)\, T(x)\, dx \quad . \tag{6}$$

Consequently, the Bayes classification procedure is to use the observation to evaluate $T(x)$; if $T(x) \geq 0$, π_1 is announced, i.e., $t(x)$ is set equal to 1. If $T(x) < 0$, we set $t(x) = 0$. This procedure minimizes the risk, with the minimum attainable risk given by

$$R(t^*,\pi) = p_1 a_1 - \int_x T^+(x)\, dx. \tag{7}$$

We have defined $T^+(x) = \max\{T(x),0\}$.

Now assume that $T(x)$ is unknown and that a training set of samples is available. The samples $\chi_n = \{X_1 \ldots X_n\}$, are assumed to be a sequence of independent observations.[1] These observations are used to form an estimate of the test function $T(x)$.

---

[1] In section 4, it will be assumed that the samples have already been correctly classified. For the present, however, no such assumption is required.

If the estimate $T_n(\chi_n,x) = T_n(X_1 \ldots X_n;x)$ is a good approximation to $T(x)$, a reasonable procedure to follow with a subsequent unclassified observation x is to let the decision function $t_n(x) = t_n(\chi_n;x) = 1$ if $T_n(x) = T_n(\chi_n;x) > 0$ and $t_n(x) = 0$ if $T_n(x) < 0$. In analogy to (6), the risk, conditioned on the training set χ_n, becomes

$$R_n(t_n(\chi_n),\pi) = p_1 a_1 - \int_x t_n(\chi_n;x)\ T(x)\ dx \quad . \tag{8}$$

$R_n(t_n(\chi_n),\pi)$, which is called the incurred risk, is a random variable since it is a function of the training set.

The unconditional risk, or overall loss, is defined to be the expectation of (8) with respect to the training set. It follows from the way in which $t_n(\chi_n;x)$ is defined, that this expectation is

$$\begin{aligned} R_n(t_n,\pi) &= E_n\{R_n(t_n(\chi_n);\ \pi)\} \\ &= p_1 a_1 - \int_x T(x)\ \Pr\{T_n(\chi_n;x) \geq 0\}\ dx \quad . \end{aligned} \tag{9}$$

The expression $\Pr\{T_n \geq 0\}$ is taken to mean "the probability that T_n is greater than or equal to zero." The two quantities of interest are the difference between the minimum risk and either (8) or (9). These are:

$$\begin{aligned} \Delta R_n(\chi_n) &= R_n(t_n(\chi_n),\pi) - R(t^*,\pi) \\ &= \int T(x)(t^*(x) - t_n(\chi_n;x))\,dx \end{aligned} \tag{10}$$

$$\begin{aligned} \Delta R_n &= R_n(t_n,\pi) - R(t^*,\pi) \\ &= \int T^+(x)\ dx - \int T(x)\ \Pr\{T_n(\chi_n;x) \geq 0\}\ dx \quad . \end{aligned} \tag{11}$$

Recall that $t^*(x)$ denotes the Bayes decision function and that $\int T(x)t^*(x)dx = \int T^+(x)dx$. Note also that $\Delta R_n = E_n\{\Delta R_n(\chi_n)\}$ and that by definition of the Bayes procedure, $\Delta R_n(\chi_n)$ is a non-negative random variable.

It is not difficult to show (e.g., [8]) that if $T_n(\chi_n;x)$ converges in probability to $T(x)$ for every x, then $\lim_{n\to\infty} \Delta R_n = 0$. Hence, as the number of training samples increases, the overall loss approaches the minimum risk. Such a process has been called asymptotically optimal by Robbins.

Van Ryzin considers the difference given by equation (10). This equation leads to the inequality ([11], Eq. (2.10))

$$0 \leq \Delta R_n(\chi_n) \leq \int |T(x) - T_n(\chi_n;x)|\,dx, \tag{12}$$

which can be used to study the convergence of $\Delta R_n(\chi_n)$. If $R_n(t_n(\chi_n),\pi)$ converges in probability to $R(t^*,\pi)$, and hence $\Delta R_n(\chi_n)$ converges to zero, the learning procedure is called Bayes risk consistent. To obtain the order of consistency, i.e., the rate of convergence, he considers dominating the expression $\int E_n|T(x) - T_n(\chi_n;x)|\,dx$.

An alternate approach is to investigate bounds on ΔR_n, the difference in overall risks. Since $\Delta R_n(\chi_n)$ is a non-negative random variable, from Markov's inequality, we have:

$$\Pr\{\Delta R_n(\chi_n) > \varepsilon\} \le \frac{1}{\varepsilon} E_n\{\Delta R_n(\chi_n)\} = \frac{1}{\varepsilon}\Delta R_n \quad . \tag{13}$$

Consequently, if the learning procedure is asymptotically optimal, it is also Bayes risk consistent.

BOUNDS ON THE RISK

Define the quantity $T^-(x) = \min\{T(x),0\}$. Equation (11) can be expressed as

$$\begin{aligned} 0 \le \Delta R_n = &\int T^+(x) \Pr\{T_n(\chi_n;x) < 0\}\, dx \\ &- \int T^-(x) \Pr\{T_n(\chi_n;x) \ge 0\}\, dx \quad . \end{aligned} \tag{14}$$

Let A be the set $A = \{x: |T(x)| \ge \varepsilon\}$, $\bar{A}$ its complement, and consider the first expression on the right-hand side of (14):

$$\begin{aligned} &\int T^+(x) \Pr\{T_n(\chi_n;x) < 0\}\, dx \\ &= \int_{\bar{A}} T^+(x) \Pr\{T_n(\chi_n;x) < 0\} dx + \int_A T^+(x) \Pr\{T_n(\chi_n;x) < 0\} dx \\ &\le \int_{\bar{A}} T^+(x)\, dx + \int T^+(x) \Pr\{(T_n(\chi_n;x) - T(x)) \le -\varepsilon\} dx \quad . \end{aligned} \tag{15}$$

A similar bound can be obtained for the second expression in (14). Combining both bounds and assuming the losses satisfy $a_1 > a_0$, we have $\int_{\bar{A}} |T(x)| dx \le a_1 \Pr\{|T(x)| \le \varepsilon\}$. Setting this equal to $\delta(\varepsilon)$, equation (14) can be dominated by

$$0 \le \Delta R_n \le \delta(\varepsilon) + \int_A |T(x)| \Pr\{|T_n(\chi_n;x) - T(x)| > \varepsilon\} dx \tag{16}$$

Suppose that the mean-square error for fixed x can be bounded by $\beta_n^2(x)$:

$$E_n\{(T_n(\chi_n;x) - T(x))^2\} \le \beta_n^2(x) \quad . \tag{17}$$

An application of the Chebyshev inequality gives

$$0 \le \Delta R_n \le \delta(\varepsilon) + \int_A |T(x)| \frac{\beta_n^2(x)}{\varepsilon}\, dx \quad . \tag{18}$$

Equations (16) and (18) can be used to obtain bounds on the risk for a variety of situations. These include supervised and and nonsupervised learning, and the situation in which the test function T(x) is partially or completely unknown. In the next section, we take the densities $f_\lambda(x)$ as the only unknowns and, with one exception, also assume that correctly classified samples are available to estimate the densities. Thus the estimate $T_n(\chi_n;x)$ can be written as $T_n(\chi_n;x) = a_1 p_1 \hat{f}_{1r}(x) - a_0 p_0 \hat{f}_{0s}(x)$, with s and r the number of samples from class π_0 and π_1, respectively, and $n = r+s$.

Before proceeding, we remark that under reasonable conditions, $\delta(\varepsilon)$ appearing in (16) and (18) can be made arbitrarily small by suitable choice of ε. One set of sufficient conditions is that $f_\lambda(x), \lambda=0,1$, be continuous and that the measure of the set $\{x: T(x)=0\}$ be zero. It is also worthwhile to note that a number of the training algorithms proposed (e.g., [2,7]) lead to an estimate $T_n(\chi_n;x)$ which converges, generally with probability one, to a particular function $T^*(x)$. Among the class of permissible approximations to $T(x)$, it is shown that $T^*(x)$ minimizes the mean-square difference $\int (T(x)-T^*(x))^2 f(x)\,dx$.[1] In contrast, the above bounds on the risk are given in terms of the mean-square error for fixed x, i.e., $E_n\{(T(x)-T_n(\chi_n;x))^2\}$.

DENSITY ESTIMATION AND RATES OF CONVERGENCE

Let the sequence of independent random variables $X_1 \ldots X_n$, represent observations drawn from one of the unknown density functions, say $f_1(x)$. The two types of estimates which we will discuss can both be put in the form

$$\hat{f}_{1n}(x) = \int K_n(x,y)\,dF_{1n}(y) \quad , \tag{19}$$

where $F_{1n}(y)$ is the empirical distribution function. If $K_n(x,y) = \frac{1}{h(n)} K((x-y)/h(n))$, the estimate is of the kernel type (see [6]):

$$\hat{f}_{1n}(x) = \frac{1}{nh(n)} \sum_{i=1}^{n} K_n((x-X_i)/h(n)) \quad . \tag{20}$$

Taking $K_n(x,y) = \sum_{j=0}^{q(n)} \gamma_n \varphi_j(x) \overset{\psi}{\varphi}_j(y)$, the estimation technique is that of the potential function or stochastic approximation method ([1,3,4]). For this case, the estimates we will discuss are

$$\hat{f}_{1n}(x) = \sum_{j=0}^{q(n)} \hat{a}_{jn} \varphi_j(x) \tag{21}$$

where q(n) is a sequence of positive integers,

$$\hat{a}_{jn} = \frac{1}{n} \sum_{i=1}^{n} \varphi_j(X_i), \tag{22}$$

and the $\varphi_j(x)$ are an appropriate set of orthogonal functions.

In order to use the bound on the risk given in (16), one has to estimate the quantity $\Pr\{|\hat{f}_{1n}(x)-f_1(x)|>\varepsilon\}$. Define: $r(n,x) = \int \left| \frac{\partial}{\partial y} K_n(x,y) \right| dy$, the bias $b(n,x)=E\hat{f}_{1n}(x)-f_1(x)$, and $\varepsilon_1=(\varepsilon-|b|)/r$. Using some results of Smirnov (see [5] for the details), one can show that

---

[1] The class of permissible approximations is generally taken to be a finite linear combination of known functions and (unknown) parameters.

$$\Pr\{|\hat{f}_{1n}(x)-f(x)|>\epsilon\}\le\Pr\{\sup_x|F_n(x)-F(x)|>\epsilon_1\}\le c_1e^{-\alpha\epsilon_1 n} \quad (23)$$

where c_1 and α are constants, with $0<\alpha\le 2$, and n is the number of observations. In order to obtain a rate of convergence, one must specify $K_n(x,y)$ and estimate the quantities $b(n,x)$ and $r(n,x)$. An easier method of bounding the risk is to estimate $\beta_n^{\,2}(x)$. For the above estimation techniques, it will be possible to obtain such a bound which is uniform in x.

With estimates of the type given in (20), one can show ([6]) that under suitable conditions on the kernel $K(y)$ and with continuity of $f(x)$, $E_n\{(\hat{f}_{1n}(x)-f_1(x))^2\}=0(1/n^{4/5})$.[1] Then, letting n_* equal the minimum of the number of learning samples from π_0 or π_1, and noting that $\int|T(x)|dx\le a_1$, equation (18) becomes

$$0\le\Delta R_n\le\delta(\epsilon)+\frac{a_1}{\epsilon}\,0(1/n_*^{\,4/5}) \quad . \quad (24)$$

Consequently, under very weak conditions on the density functions, one can obtain convergence at the rate given by (24).

For the second estimation technique (Eqs. (21)-(22)), and with no knowledge of the densities $f_\lambda(x)$, it is natural to pick the orthogonal functions as the Hermite functions since they are orthogonal on the real line. Choosing these functions, we have the following result.

Theorem ([9]):

Assume $f(x)$ is continuous, of bounded variation, $L_1\cap L_2$ in $(-\infty,\infty)$. Furthermore, assume $\exp(x^2/2)(d^r/dx^r)[f(x)\exp(-x^2/2)]$, for $r\ge 3$, is square integrable. Then, with $q(n)=0(n^{1/r})$, the mean-square error satisfies $E_n\{(f_n(x)-f(x))^2\}=0(1/n^{(r-2)/r})$.

Consequently, the difference in risks can be dominated by

$$0\le\Delta R_n\le\delta(\epsilon)+\frac{a_1}{\epsilon}\,0(1/n_*^{\,(r-2)/r}). \quad (25)$$

Again, n_* is the minimum number of samples from π_0 or π_1.

If the densities $f_\lambda(x)$ can each be represented by a finite expansion of the above form (not necessarily in terms of the Hermite functions), then one does not need correctly classified observations in order to converge to the optimum test function ([10]). Hence, the "training period" never terminates - one is classifying observations and updating the estimate of $T(x)$ simultaneously. For this happenstance, convergence of the risk takes place at the rate $0(1/n)$, where n is now the total number of observations. We observe that other rates can be obtained, both for this case and for the second estimation technique (equation (25)), by estimating the quantities $\Pr\{|\hat{a}_{jn}-a_j|\ge\epsilon\}$ and using (16) to bound the risks (see [12]). However, these results give only a rate of convergence; the present results, with some assumptions on the (unknown) densities, lead to bounds as well as rates (e.g., [9]).

---

[1] By $t_n(x)=0(1/n^\delta)$ we mean there exists a constant c such that $t_n(x)\le c/n^\delta$, for all x.

CONCLUSIONS

Bounds on the difference between the risk in classifying an observation after the learning period has terminated and the minimum Bayes risk have been derived (Eqs. (16),(18)). Rates of convergence have been given (Eqs. (24),(25)) for the class of learning procedures which estimate the unknown Bayes test function by estimating the underlying probability densities associated with each class.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under Grant NSF-GK-1532.

REFERENCES

1. Aizerman, M., Braverman, E., and Rozonoer, L., "The Probability Problem of Pattern Recognition Learning and the Method of Potential Functions," Automation and Remote Control, Vol. 25, 1965, pp. 1175-1190.

2. Blaydon, C. and Ho, Y. C., "On the Abstraction Problem in Pattern Classification," Proceedings of the National Elect. Conf., Vol. 22, 1966, pp. 857-862.

3. Čencov, N. N., "Estimation of an Unknown Distribution Density from Observations," Soviet Mathematics, Vol. 3, 1962, pp. 1559-1562.

4. Kashyap, R. L., Blaydon, C.C., "Recovery of Functions from Noisy Measurements Taken at Randomly Selected Points and Its Application to Pattern Classification," Proc. IEEE, Vol. 54, 1966, pp. 1127-1129.

5. Nadaraya, E. A., "On Non-Parametric Estimates of Density Functions and Regression Curves," Theory of Probability and Its Applications, Vol. 10, 1965, pp. 186-190.

6. Parzen, E., "On the Estimation of a Probability Density Function," Ann. Math. Stat., Vol. 33, pp. 1065-1076.

7. Patterson, J. D., Wagner, T. J., and Womack, B. F., "A Mean-Square Performance Criterion for Adaptive Pattern Classification Systems," IEEE Trans. on Automatic Control, Vol. 12, 1967, pp. 195-197.

8. Robbins, H., "The Empirical Bayes Approach to Testing Statistical Hypotheses," Review of the International Statistical Institute, Vol. 31, 1963, pp. 195-208.

9. Schwartz, S., "Estimation of Probability Density by an Orthogonal Series," Ann. Math. Stat., Vol. 38, 1967, pp. 1261-1265.

10. -----, "An Example of Nonsupervised Adaptive Pattern Recognition," IEEE Trans. on Automatic Control, To appear, Vol. 13, No. 1.

11. Van Ryzin, J., "Bayes Risk Consistency of Classification Procedures Using Density Estimation," Sankhya, A, Vol. 28, 1966.

12. Wagner, T. J., "The Rate of Convergence of an Adaptive Pattern Classification Scheme," Proc. Princeton Conf. on Info. Sciences and Systems, 1967, pp. 95-97.

EMPIRICAL DECOMPOSITION OF MIXTURES IN PATTERN RECOGNITIONS

T. L. STEWART AND W. M. HAMMOND
Dept. of Electrical Engineering, Bradley University, Peoria, Illinois

Abstract: An important problem in pattern recognition or signal detection is the construction of a detector to operate in an environment that is incompletely described statistically. The ultimate objective is to construct a detector that approximates the optimum (minimum probability of error) decision rule given by: decide i^{th} class active for i which maximizes

$$P(\omega_i \mid X) = \frac{p(X \mid \omega_i)\, P(\omega_i)}{p(X)} \tag{1}$$

where ω_i indicates the i^{th} pattern class and X the noisy measurement on which the decision is based. In this paper, we assume the noise is additive of known functional form. The various quantities required to define (1) are unknown (i.e., the a priori probabilities $P(\omega_i)$, the parameters which characterize $p(X|\omega_i)$, possibly the functional form of $p(X|\omega_i)$). We have previously received an unclassified set of samples $\{X_i\}$.

With the noise density $p_N(\cdot)$ known, and assuming the signal and noise are independent, the density on X is given by

$$p_X(\cdot) = p_N(\cdot) * p_S(\cdot) \tag{2}$$

where * denotes convolution. The signal density $p_S(\cdot)$ is called the mixing density function and is given by

$$p_S(\cdot) = \sum_{i=1}^{M} p_i(S \mid \omega_i)\, P(\omega_i) \tag{3}$$

where $p_i(S \mid \omega_i)$ is the density associated with pattern class ω_i (i.e., if each pattern is determined by a constant S_i, then $p_i(S \mid \omega_i)$ is the impulse, $U_o(S - S_i)$).

The approach in this paper is to approximate the mixing density function $p_S(\cdot)$ by a series

$$p_S(\cdot) = \sum_{i=1}^{N} a_i\, \phi_i(\cdot) \tag{4}$$

where the $\phi_i(\cdot)$ are gaussian functions with different means. The mean and variance are chosen to give a satisfactory empirical representation of $p_S(\cdot)$. Given this representation of $p_S(\cdot)$, one can decompose (3) into the component densities. The coefficients a_i are chosen to minimize the integral-square error

$$I = \int_{-\infty}^{\infty} (p_S(s) - \sum_{i=1}^{N} \phi_i(s))^2 \, ds \tag{5}$$

Estimates of a_i are to be made from the set of unclassified samples $\{X_i\}$ containing signal plus noise.

No unbiased estimates for the coefficients a_i were obtained; however, a suitable estimation procedure was to use as estimates the coefficients $\hat{a}_i$ which minimize

$$I = \int_{-\infty}^{\infty} (p_X(x) - \sum_{i=1}^{N} \hat{a}_i \hat{\phi}_i(x))^2 \, dx \tag{6}$$

where

$$\hat{\phi}_i(\cdot) = p_n(\cdot) * \phi_i(\cdot) \tag{7}$$

For the examples considered, the bias in a_i was small. An algorithm for computing estimates of a_i was obtained and convergence was proved using several well-known stochastic approximation theorems.

Of particular importance, this approach has permitted the decomposition of mixtures for which the number of unknowns is too large to use the Bayes' approach of other investigators.(1,2) In addition, we require less a priori information than other approaches.

---

[1] E. A. Patrick and J. C. Hancock, "Nonsupervised Sequential Classification and Recognition of Patterns," IEEE Trans. on Info. Theory, Vol. IT-12, pp. 362-372, July, 1966.

[2] S. C. Fralick, "Learning to Recognize Patterns Without a Teacher," IEEE Trans. on Info. Theory, Vol. IT-13, pp. 57-64, January, 1967.

COMPUTER RECOGNITION OF ALPHABETIC CHARACTERS THROUGH DECOMPOSITION INTO BASIC STROKES

THEODOSIOS PAVLIDIS
Princeton University
Princeton, New Jersey

The detection of certain basic strokes is one of the many approaches used for the recognition of alphabetic symbols by computer.[1-3] These techniques often use information about the stroke sequence (which is not always available), scanning of the boundary, random search, fitting segments into a pattern, etc. The last three approaches may be very sensitive to noise. This paper presents a method which tries to overcome these difficulties.

It was shown that it is possible to decompose any polygon into certain convex subsets in a unique way and then use a Hausdorf metric to approximate these by members of a finite set of convex sets which may be thought as the basic strokes. This set can be determined through a training or learning procedure. The implementation of this method on a general purpose computer is very difficult and an approximate technique has been developed which has given satisfactory results during experimental simulation.

It is based on the observation that rectangles with sides parallel to a finite number of directions characterized by angles $\phi_1, \phi_2, \ldots \phi_m$ can form such a basis when alphabetic characters are considered.[2]

Let A be a bounded subset of the x-y plane and let $a_n^1(c)$, $a_n^2(c), \ldots . a_n^r(c)$ be the points at which is intersected by a line with equation

$$\cos\phi_n . y - \sin\phi_n . x = c$$

If a positive direction has been defined on the line one may consider the first two points of the intersection and denote the length of the linear segment between the two by $s_n^+(c)$. This function will be called a (positive) integral projection in the direction of ϕ_n.

It is obvious that any peak of the integral projection will correspond to a subset of A containing linear segments in the corresponding direction and, therefore, such a segment could be approximated with a rectangle whose greater side is parallel to this direction. This observation forms the basis for an algorithm determining the basic strokes contained in A. Once a stroke is determined a corresponding part of the character is removed and the process is repeated until no further decomposition can be obtained.

What makes this process interesting is that one can obtain good results when dealing with typewritten characters by using only three directions, $\phi_1 = 0^\circ$, $\phi_2 = 45^\circ$ and $\phi_3 = 90^\circ$, and for each one of them three or four rectangles only.

As a matter of fact, one can design the algorithm by considering a very limited number of samples (one or two from each font). Essentially it implements the intuitive characterizations of the letters of the alphabet. A larger training or design set of samples was subsequently used only for the determination of certain numerical parameters involved.

The decision rule used was the following: The range of the parameter c was divided in intervals and on each one of them a set of logical variables (one for each basic stroke) were defined which would have the value 1 if the corresponding stroke occurred for that value and zero otherwise. For each class a Boolean function is defined which has the value 1 if, and only if, the character considered belongs to this class. (In contrast with earlier methods, it need not be threshold realizable).

Tests of the method were carried for multifont lower case typewritten characters stored on a video tape*. In a 3 letter alphabet (ℓ, f and t) no errors were made on the testing set (619 samples). In a 10 letter alphabet the error rate was 1.63% on the testing set (2267 samples). In both cases no errors were made on the design sets.

REFERENCES

1. Mermelstein, P. and M. Eden, Information and Control, vol. 7, (1964), pp. 255-270.

2. Spinrad, R.J., Information and Control, vol. 8, (1965), pp. 124-142.

3. Muchnik, I.B., Automation and Remote Control, vol. 27, No. 10, (October 1966, transl. April 1967), pp. 1737-1747.

ACKNOWLEDGEMENT: This research was partly supported by NSF Grant GK-1630.

---

* This was supplied by the IBM T. J. Watson Research Center.

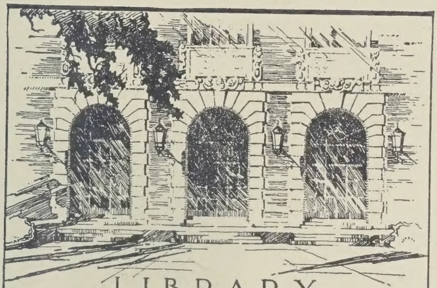
INDEX OF AUTHORS

AGGARWAL, J. K. 451
ALLEN, R. H. 534
AOKI, M. 191
ATHANS, M. 614
BAKER, W. D. 557
BARBAY, J. E. 730
BEHN, R. D. 317
BHATTACHARYYA, B. B. 263
BICKART, T. A. 232
BITTEL, R. 596
BOSE, N. K. 461
CHANG, F. Y. 290
CHEN, C. F. 228,231
CHEN, W. K. 300
CHIEN, Y. T. 337
CHO, Y. S. 238,249
CHYUNG, D. K. 222
CLELAND, L. L. 692
COHEN, G. H. 104
CRETELLA, J. P. 39
CROSS, E. R. 152
D'ANGELO, H. 142,711
DAVIDSON, E. S. 683
DAVILA, E. A. 402
DERTOUZOS, M. L. 384
DESOER, C. A. 8
DOMPE, R. J. 170
DORF, R. C. 170
EPSTEIN, M. I. 97
FINN, H. M. 502
FU, K. S. 47,337,374
FU, Y. 729
GIGUERE, J. C. 263
GITLITZ, C. R. 608
GRAMMATICOS, A. 97
HACHTEL, G. D. 564
HADDAD, A. H. 770
HAMMOND, W. M. 808
HELLER, J. E. 428
HENDRICKS, T. 142
HINDIN, H. J. 28
HO, Y. C. 317
HSIA, T. C. 211
HSIAO, M. Y. 67
IQLEEM, M. J. 201
ISHIZAKI, Y. 644
KAHNE, S. J. 161
KAJITANI, Y. 635,644
KARNI, S. 139
KASHYAP, R. L. 501
KINARIWALA, B. 4
KISHI, G. 635,644

KLEINMAN, D. L. 614
KOBAYASHI, H. 491
KOIVUNIEMI, A. J. 459
KOKOTOVIĆ, P. 428,577
KRUH, P. 459
KUO, B. C. 213
KURZ, L. 364
LAMONT, G. B. 161
LaPATRA, J. W. 738
LAVI, A. 575
LAYTON, D. M. 781
LEAKE, R. J. 57
LEE, S. C. 119,279
LEON, B. J. 692
LIN, P. M. 627
LIU, B. 701
LUH, J.Y.S. 152
LUM, V. Y. 482
LUPO, F. J. 748
McCLAMROCH, N. H. 451
McDANIEL, W. L., Jr. 791
MANCUSO, T. G. 119
MARSHALL, T. G., Jr. 544
MARTENS, G. O. 739
MASTEN, M. K. 181
MAYEDA, W. 671
MEDANIC, J. 439
MEIER, L. 721
METZE, G. 683
MEYERHOFF, D. E. 513
MICHEL, A. N. 230
MONOPOLI, R. V. 608
MOWLE, F. J. 77
MURRAY-LASSO, M. A. 557
MUSA, S. A. 229
MYERS, B. R. 402
NARASIMHAN, T. R. 460
NARAYANAN, S. 575
NARENDRA, K. S. 238,249
NIKOLIC, Z. J. 374
OHTSUKI, T. 644
OSTRANDER, L. E. 104
PAPOULIS, A. 87
PATRICK, L. B. 711
PAVLIDIS, T. 810
PENBECI, S. S. 279
POTTLE, C. 523,534
PRADA, G. 29
RAMACHANDRAN, V. 460
REDDY, D. C. 12
RESH, J. A. 112
REZA, F. M. 461
ROHRER, R. A. 757
ROSKO, J. S. 572
SAIN, M. K. 327
SALMON, D. M. 577
SANDBERG, I. W. 1
SANNUTI, P. 428

SARIDIS, G. N. 374
SCHLICHT, D. L. 586
SCHMIDLIN, D. J. 748
SCHUMER, M. A. 346
SCHUMPERT, J. M. 792
SCHWARTZ, S. C. 800
SHAFFER, C. V. 269
SHEN, D.W.C. 228
SHIEH, L. S. 231
SIBUL, L. H. 356
SILJAK, D. 596
SINHA, N. K. 201
SOUZA, C. 57
STEELMAN, J. E. 139
STEIGLITZ, K. 346
STEWART, T. L. 808
STRAKHOV, N. A. 364
SU, J.C.C. 573
SWAMY, M.N.S. 263
TABAK, D. 213
TAHIM, G. S. 259
TANG, D. T. 482
THALER, G. J. 586
THOMAS, J. B. 491
TSUCHIYA, T. 644
VAN VALKENBURG, M. E. 701
VIMOLVANICH, V. 211
WATANABE, H. 644
WEE, W. G. 47
WEINBERG, L. 654
WERNER, R. A. 418
WILLIS, K. D. 738
WING, O. 290
WITSENHAUSEN, H. S. 19
WOHLERS, M. 412
WOMACK, B. F. 181
WU, S. H. 230
WUJEK, J. H., Jr. 129
YAU, S. S. 792
ZEHEB, E. 470
ZOBRIST, G. W. 730

[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]
[illegible] [illegible]



LIBRARY
OF THE
UNIVERSITY
OF ILLINOIS

NOTICE: According to Sec. 19 (a) of the University *Statutes,* all books and other library materials acquired in any manner by the University belong to the University Library. When this item is no longer needed by the department, it should be returned to the Acquisition Department, University Library.

b

UNIVERSITY OF ILLINOIS-URBANA
621.31921AL54P C003
PROCEEDINGS.$URBANA ILL
5 1967
3 0112 006279621